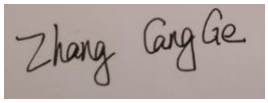
A novel Transformer pre-training objective and a novel fine-tuning method for abstractive summarization

CangGe Zhang
17983554
Auckland University of Technology
sms8171@autuni.ac.nz

Content

Attestation of Authorship

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person except that which appears in the citations and acknowledgments. Nor does it contain material which to a substantial extent I have submitted for the qualification for any other degree of another university or other institution of higher learning.

Signature:  Date:15/02/2022

Acknowledgment

I would like to show my sincere appreciation to my supervisor Parma Nand. His patience and brilliant suggestions helped me finish this wonderful project. He helped me to be an eligible effective researcher that who could design experiments and write a thesis for scientific requirement.

I also like to show my appreciation to my family. Their patience, support and understanding helped my me pass evade difficulties along the way.

CangGe Zhang

Auckland, New Zealand

8 February 2022

List of Figures

List of Tables

Abstract

Pre-training Transformer has been widely used in many NLP tasks including document summarization. Researchers designed many different self-supervised objectives for their pre-training transformer models, then based on the seq2seq model to fine tune on these pre-trained Transformer models for downstream tasks. However, most researchers designed their self-supervised objectives for all NLP tasks, the ability of self-supervised objectives for a specific task such as abstractive document summary hasn't been largely explored. This article designed a novel self-supervised objective MSLM (Mask Summary Language Model) for document summarization. MSLM uses labeled document summary corpus for pre-training, where some words have been removed/masked from the summary. The source text concatenates the masked summary as the input, while the output is the summary with the original words masked. The objective is to predict the masked words from the summary. We first pre-trained on three variants of MSLM that remove nouns, verbs, and all the other words from the summary respectively. We found that removing nouns from the summary obtained the best ROUGE score on the downstream abstractive document summarization task. Then, inspired by BERT (Devlin et al., 2018) and Roberta (Liu et al., 2019), we pre-trained the concatenation of MLM (Mask Language Model that first been proposed in BERT) and our best MSLM variant, we found that fine-tuning the model that pre-trained on the concatenation of MLM and MSLM obtained higher ROUGE score than the model that pre-trained on MLM only.

In addition to the new way of training, we also designed a novel fine-tuning method - Diverse Aspects Document Understanding (DADU). When human mothers communicate with their children, they sometimes emphasize nouns, sometime emphasize verbs (CHOI, 2000). We assume that through such different emphasis training, children could obtain many different views for same scene, which is then merged to help in comprehension. Based on this assumption, we split the summary generation task into three sub-tasks; each task was trained on a seq2seq model for the same article. The target of these three sub-tasks are different, corresponding to the three different structures for the article. The three targets focused on nouns, verbs, and all the other words except nouns and verbs respectively. Then we used a merge function to generate the final summary. Compared to the state-of-art fine-tuning methods, DADU obtains a higher score of precision on rouge-1 and rouge-3, but lower on rouge-2.

Chapter 1 Introduction

This chapter will introduce the motivation and inspiration of this thesis. We will show the general architecture of our model and method. We also conclude the main contributions in our thesis.

1.1 Introduction

The aim of abstractive document summary is to generate summaries that might use novel words and novel sentences that do not exist in the source text. In contrast, extractive document summarization only extracted important sentences from the original document. In recent years, abstractive document summary results highly improved with Transform (Vaswani et al, 2017) and sequence-to-sequence (Sutskever et al, 2014) technology. Researchers usually designed some self-supervised objectives for Transform pre-training. After pre-training, researchers could obtain a better representation for input documents. Sequence-to-sequence model has been used as fine-tuning that will load the pre-training Transformer model as encoder and decoder to finish the downstream task such as abstractive document summary.

BERT first proposed Masked Language Model (MLM) objective and Next Sentence Prediction (NSP) objective. After BERT, many researchers proposed BERT-like models. For example, Roberta evaluated the contribution of MLM and NSP for downstream tasks. They reported that NSP doesn't provide a significant contribution for downstream tasks, thus they only kept MLM objective when pre-training Roberta (Liu et al, 2019). BART adds Random Sentence objective when pre-training their model. Based on their work, the ROUGE score for the abstractive document summary task was highly improved. However, all these BERT-like models are designed for all NLP tasks such as document classification, translation, text understanding. Based on our knowledge, only one research group proposed PEGASUS (Zhang et al, 2020) tailored the objective only for document summary, and obtained a state-of-art ROUGE score.

In this article, we designed a novel Transformer pre-training objective (MSLM) and a novel fine-tuning method (DADU) for document summary especially. The inspiration of both MSLM and DADU comes from the practice that we did when we were learning a language in school. The practice usually will let students read an article first, then show the summary with some blanks to students, and ask students to finish those blanks to make the summary for correct and fluent sentences. After repeated practice, students will get the ability to summarize documents themselves. That is the reason we chose pretraining on a supervised dataset, not like other objectives such as MLM who did self-supervised training on a non-supervised dataset.

We named our novel objective Mask Summary Language Model (MSLM) and designed three variants of MSLM (MSLM-noun, MSLM-verb, MSLM-rest). As the name represents, MSLM-noun only masks noun words in the summary, MSLM-verb only masks verb words, and MSLM-rest masks those words are not nouns or verbs. In our work, we pre-trained three MSLM variants on CNN/daily-mail datasets and fine-tuned based on a traditional seq2seq model (Sutskever et al, 2014) for abstractive

document summary task on two datasets: Food Review and CNN/daily-mail. We found that MSLM-none got the highest score. We also re-pre-trained Roberta on CNN/daily-mail (original Roberta pre-training on Weki-Pedia dataset). We found that MSLM-noun got a lower score than MLM for the downstream document summarization task. Then we concatenated on MLM and MSLM-noun, we found MLM & MSLM got higher ROUGE scores than only using MLM.

We named our novel fine-tuning method Diverse Aspects Document Understanding (DADU). Compared to the traditional fine-tuning method, DADU does not pursue understanding all aspects of a document in one task. In our work, we split fine-tuning document summary into three sub-tasks: DADU-noun, DADU-verb, DADU-rest. The architecture of these three DADU sub-tasks is the same (a simple seq2seq model). However, for the same article, the targets of these three tasks are different. The targets are preprocessed summaries focusing on nouns, verbs, and all the other kinds of words respectively. Three different version summaries for one same article could be generated based on these three sub-tasks. We implemented a merge function to merge the results from the three sub-tasks to create the whole summary.

Pre-training and fine-tuning architecture are shown in Figure 1 and Figure 2.

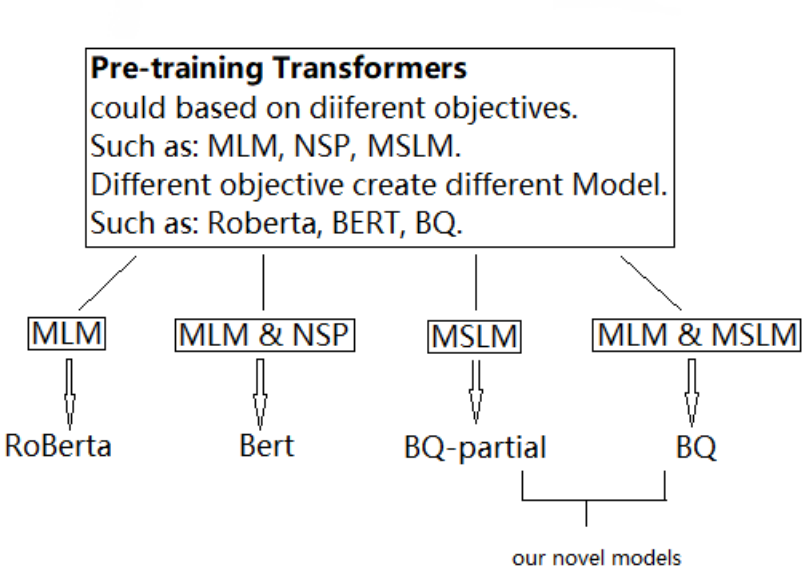


Figure 1: Models and objectives.
The Transformer model pre-trained on MLM objective named Roberta. The model pre-trained on MLM and NSP was named Bert. The model we proposed in this thesis BQ pre-trained on MSLM only and MLM concatenate MSLM.

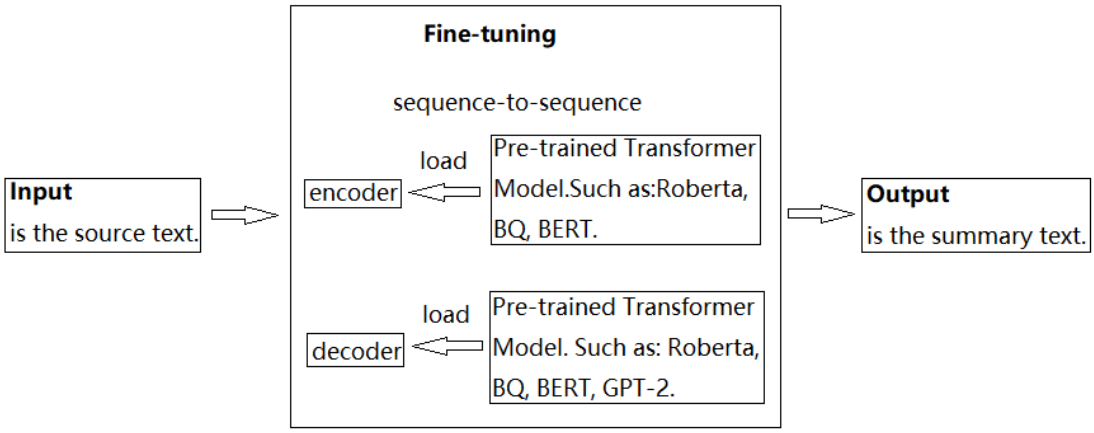


Figure 2: Fine-tuning architecture

Fine-tuning base on sequence-to-sequence (SEQ2SEQ) model. SEQ2SEQ includes two parts, encoder, and decoder. The pre-trained Transformer model will be loaded into encoder and decoder, and the encoder and decoder could load different models. For example, the encoder could load BERT, the decoder could load GPT-2.

1.2 Research Questions

The main two research questions of this article as below:

Does the new objective MSLM has contribution for pre-training model?

Does the new fine-tuning method DADU could generate state-of-art rouge score abstractive summary?

1.3 Contributions

To summarize our contribution:

We proposed a novel objective (MSLM) and pre-trained our model (BQ) based on MSLM. We evaluated the performance of three variants of MSLM on the document summarization task.

We evaluated the performance of our model (BQ) that pre-trained on the objective (MSLM_noun) and Roberta on the downstream abstractive document summary task.

We concatenated MLM and MSLM (MLM & MSLM) objectives and evaluated them on downstream abstractive document summary tasks. Compared to only using MLM, MLM & MSLM got a higher F1 ROUGE score.

We proposed a novel fine-tuning Method (DADU) and evaluated it on CNN/daily-mail abstractive document summary task. Compared to the traditional fine-tuning method, DADU got a higher precision score for Rouge-1, Rouge-L.

1.4 Section Introduction

The rest of this thesis includes five chapters. Chapter 2 focuses on literature review; we introduced the background knowledge of abstractive document summary first. Then we introduced the Transformer model and fine-tuning method, besides that we also introduced other researchers' models that are based on the transformer model and fine-tuning method and their achievement. We also introduced a new research direction of abstractive summarization that includes evaluating the factuality, error correction, and improving the salient and entailment for the abstractive generated summary. Chapter 3 introduced the novel objective (MSLM) and its' three variants, and the novel fine-tuning method (DADU) that was proposed in this thesis. Chapter 4 introduced our three experiments and their evaluation method ROUGE first. Then we introduced the datasets we used in our experiments and the data pre-processing method. After that, we introduced the implementation of our experiments. In the end, we showed the results of our three experiments and evaluated the results. We also discussed the limitations of our experiments. Chapter 5 presents the conclusion of our work and discusses our future work.

Chapter 2 Literature Review

In this chapter, we introduced other researchers' works on abstractive document summary, transformer model, and the fine-tuning method through literature review. Through review their work could show the key conceptions in our work clearly. And it also could clearly show the contribution of transformer model, and fine-tuning method for abstractive document summary task. We also introduced a new research direction that emerged in recent years with the progress of abstractive summary. This direction included new metrics that have been used to evaluate the factuality of generated summary method to improve the coherence and correctness of generated summary.

2.1 Introduction

Document summarization is the task of creating concise and condensed version of the source text. Extractive document summarization is a relatively early method that is shown to solve this task. Extractive document summarization usually will create some method to extract important sentences in the source text, then reuses these sentences to create a summary of the document. Such as: (Kupiec et al, 1995; Conroy and O'leary, 2001) treat this task as a sentence ranking problem, they classify and score the sentences then decide which to use to create new summary. Nal- lapati et al. (2017) proposed SUMMARUNNER is one of the earliest works available which applied neural approaches to solve this problem. (Zhang et al, 2019b; Liu and Lapata, 2019) employed pre-training Transformers in their extractive summarization model.

With the advancement and development of NLP and Artificial intelligence researchers do not believe in just reusing the original sentences in source text, researchers hope their models would generate new sentences using novel words that are not shown in source texts to generate summaries like human beings. For this purpose, abstractive document summarization has received more attention. For examples: Rush et al (2015) and Nallapati et al. (2016) employed SEQ2SEQ LSTM architecture to solve text summarization. Gu et al. (2016) extended their work with copy mechanism that could copy words from source texts and (See et al, 2017) extended their work with coverage mechanism that could keep track of the words in summary. Bottom Up (Gehrmann et al, 2018) combined words extraction and abstractive summary generation. Their model uses bidirectional LSTM model to train on words selection objective that determines which word should be included in the summary. Then they fused the words selection distribution into a decoder to generate a summary. The words selection distribution indicates the next word should copy from original or generate. Liu and Lapata (2019) loaded a pre-trained Transformer model (BERT; Devlin et al. (2019)) to their SEQ2SEQ transformer encoder. Fabbri et al. (2019) incorporated extractive model and single document summarization model to create a multi-news abstractive summarization model. Besides that, they first proposed a dataset (MDS) for multi-documents summarization task. They used a point generator network (See et al, 2017) to extract information from source text and use Maximal Marginal Relevance (MMR) (Carbonell & Goldstein, 1998) score to ranking sentence.

Chen and Bansal (2018) proposed a fast abstractive summarization model that selects the salient sentences from source text, then rewrote these sentences. They employed two neural models to conduct these training respectively, and since they conducted these two trainings parallel, the convergence speed was 4 times faster than previous models. Since they extract important sentences first, their model avoids redundancy of generated summary that are commonly found in other models They employed a temporal convolutional model to represent sentences and applied another bidirectional LSTM-RNN on the output of convolutional model to add context information to the

sentence representation. Based on the sentence representations, they trained a point-network (Vinyals et al, 2015) to extract salient sentences.

The rest of this chapter will focus on Language Model, Pre-training Transformer, the methods that are used to generate text. At the end, we will introduce new metrics that are used to evaluate the faithfulness of the generated summaries, and the methods to improve the salient and entailment of generated summaries.

2.2 Language Model

Al-Rfou et al (2018) designed a character level language model that uses Transformer architecture instead of RNNs and they designed a set of losses to improve performance. They add prediction at intermediate layer and added extra prediction for every position at final layer to speed up convergence. All losses of additional predictions were added to the total loss with discounted weight. Their model outperformed LSTMs and they proved Transformer architecture could improve the language modeling. However, they separated the sequence into a few fixed length segments that caused the model to lose contextual information.

Dai and Le et al (2015) first proposed language model (LM) fine-tune method, they trained on recurrent network that used unlabeled data with two tasks to improve sequence representation. 1) predict next words in a sequence 2) autoencoder a sequence input, then predict the sequence again. They named the process that train model on these two tasks as ‘pretraining’, other downstream tasks such as classification, translation could fine-tune on their model.
Vaswani et al (2017) proposed Transformers which highly improved results of many NLP tasks included abstractive document summary. Pre-training Transformers on large corpus then fine-tuning on specific task has become ubiquitous in NLP. Different from MLM mask single word, MASS (Song et al, 2019) randomly masked fragments in sentences, using remaining parts of sentences to predict the original text. UniLM (Dong et al, 2019) combined using three types of objectives: masked, unidirectional and SEQ2SEQ to pre-training. The objective of GPT-2 (Radford et al, 2019) is to predict the next word in sequence. BART (Lewis et al, 2019) corrupted the source text into random segments and shuffled them, then let their model reorganize the segments to their original order. For mask, BART used a single [MASK] symbol mask to a span of words. Pretraining Transformers was limited by fixed length context, Transformer-XL (Dai et al, 2019) based on relative position encoding and considering previous output proposed a large-scale Transformer model. Their model obtained strong perplexity and could generate coherent text. This was different to most abstractive summarization approaches that are based on encoder-decoder recurrent neural networks, Narayan et al (2018a) introduced a convolutional neural network-based model that particularly suited single sentence summaries. Rothe et al (2020) performed extensive experiments that set different Transformer model

checkpoints combination to seq2seq model. Their aim was to find which combination is the best for sequence generation. They described that the performance of BERT and GPT-2 combination often lowered than initialize weights randomly, RoBerta and GPT-2 combination achieved strong results.

2.3 Generate Summary Text

Paulus et al (2017) proposed a new prediction method that combined with reinforcement learning. Their model records the attention weights of previous input tokens in encoder, and record attention weights of previous predicted words in decoder. They used these two attention weights and current hidden state to predict new words. They were the first to apply end-to-end sequence model to New York Times dataset (NYT), and they reached 41.16 ROUGE score on CNN/Daily Mail.
See et al (2019) shown that massively pretrained language model GPT2-117 is better on generate content-rich text (i.e. contain more rare words), however it still has issues such as the trend to use words and sentence repeatedly and lack long range coherence.
Welleck et al (2020) proposed an unlikelihood pretraining objective that improves repetitive issues. Unlikelihood objectives correct the token use bias and penalize the tokens repeated.

2.4 Evaluate faithfulness of generated summary

Based on weakly-supervised model, Krysci ´ nski et al (2019b) proposed a method to verify factual consistency for generated abstractive summary. Their model trained on three tasks: 1) transformed sentence factuality identify, 2) extract a span of tokens from source text that is consistent with generated summary, 3) extract a span of tokens from source text that is inconsistent with generated summary.
(Wang et al, 2020) proposed a question-answering based automatic evaluation metrics - QAGS - to examine faithfulness for abstractive document summarization. QAGS will generate questions for source text, then using exact match as similar metrics to compare answers from source text and generated summary.
Maynez et al (2020) proceeded with a large-scale human evaluation on seven abstractive summarization models. They concluded that faithfulness and factual is the critical problem for abstractive document summarization, they claimed evaluation metrics ROUGE and BERTscore are not sufficient to evaluate faithful and factual of abstractive summary, they found semantic inference-based automatic measures are better for summary quality evaluation.

Falke et al, (2019) build a crowdsourcing evaluation based on human evaluation and argued that human evaluation is the only reliable method to evaluate faithfulness of generated summaries. They let the workers label each sentence

(correct or incorrect for source text) in summaries. They assigned at least two workers for the same summary and collected their labels and used a Bayesian model to merge the results. They argued that the entailment of summary should be supported by the source text, thus they tried to use NLI (Natural Language model) to re-rank the generated summaries to improve the correctness of summaries. Their experiments are based on FAS abstractive summarization model and use different windows size of beam search to generate different version of summary. Then they applied NLI model (i.e. ESIM) on generated summaries and re-rank summaries by results. Then they evaluated the summary that obtain the highest NLI prediction score by crowdsource method. They found that highest NLI prediction score does have a strong positive relation with factuality of generated summary.

Goyal & Durrett (2020) assumed that the dependency of each subject and objective pair in generate summary should also be found in source text.

2.5 Improve faithfulness of generated summary

Pasunuru & Bansal (2018) improved coherence of abstractive summarization as a reinforcement multi-task. They proposed ROUGESal and Entail reinforce reward and took them into account during training. This was different to ROUGE who treated every token equal, ROUGESal gave higher weight to salient words/phrases. They trained a classifier for calculating entailment score on SNLI (Bowman et al, 2015) and Multi-NLI (Williams et al, 2017) dataset. Then, they used average entailment score of reference summary and generated summary as reword.

Pasunuru et al (2017) fused entailment generation into abstractive summary generation by multi-task learning. They built entailment generation model and summary generation model separately and employed sequence-to-sequence multi-task learning (Luong et al, 2016) to share decoder parameters that generated summary to entailment generation model. For summary generation model they used a two layered LSTM-RNN to encode source text and used a two layered LSTM-RNN to decode when generating summary. For entailment generation models they used the same architecture. They set Gigaword Corpus as input to summary generation model and SNLI corpus that provided entail-labeled pairs as input to entailment generation model. For Gigaword Corpus, they chose the first sentence as the source text and the headline as the target text. They evaluated by ROUGE score, and they proved that even the training entailment generation model on different information domain (i.e. SNLI corpus), the multi-task learning model still could improve the abstractive summarization (Luong et al, 2016).

(Guo et al, 2019) used two auxiliary models: question answering model and entailment generation model to improve information salient and entailment of generated summary. They also proposed a novel architecture for multi-task learning that could settle different parameter sharing policy for different layers. For question generation model they choose SQuAD dataset and same as (Du et al, 2017) method to identify the salient information for a given sentence first, then generated questions and answers for the salient information. For entailment generation they also used entailment-labeled pair in SNLI dataset as (Pasunuru et al, 2017). And they treated entailment generation as an RTE classification task. They used two layers bidirectional LSTM-RNN for encoder and other two layers bidirectional LSTM-RNN for decoder. And they shared the parameters of three tasks at second layer of encoder and first layer of decoder. They proved that abstractive summary model incorporates either Question Generation Model or Entailment Generation Model singly could improve the saliency and entailment of summary. And by incorporating the two auxiliary models with abstractive summary model simultaneously reached the best ROUGE score on CNN/Daily Mail Dataset and Gigaword datasets.

(Cao et al, 2018) extracted facts from each sentence by mature task Open Information Extraction (Banko et al, 2007). The extracted facts represented by a tuple (subject, relation, object), and they used every tuple by creating a concise sentence and named it as fact description. All fact descriptions have been added as an extra input to the source text. Their results showed that the generated summaries are 40% more likely to use the words in extracted fact description. Since not every tuple is complete, they employed dependency parser to merge related incomplete tuples into one fact description. They used bidirectional Gated Recurrent Unit (BiGRU) to encode source text and fact description, respectively. They built a dual-attention GRU to decode. At each time step t, decoder will generate context vector Cs for source text and Cr for fact description. (Cao et al, 2018) proposed two approaches to combine the two context vectors. The first is concatenate the two vectors directed, the second is combine context vectors with weight sum.

(Li et al, 2018) incorporated knowledge into abstractive summarization. They also used multi-task learning and built the entailment-aware encoder, they also fused the entailment-aware into decoder by Reward Augmented Maximum Likelihood (RAML) training (Norouzi et al, 2016). They employed a BiLSTM to encode, and for entailment-aware, the labeled entailment sentence pair was fed into an encoder to generate two vectors; r u and v respectively. They fed the concatenation of the absolute difference and the element-wise product of u and v to a multilayer perceptron (MLP) classifier. In equation (1, u is the entailment sentence vector, v is the label vector.

$$q = [|u - v|; u * v; u; v] \qquad (1$$

Different from previous work, they considered abstractive summarization to be the main task, they set different weights for abstractive summarization and entailment-aware model. Such as training abstractive summarization 100 batches then training entailment-aware 10 batches. For decoder RAML training, it sampled the input first, the generated vector of sampling was used as reward. For a given sentence with length L, they counted the number of sentences in a range from 0 to 2L, then weight counts and performed a normalization on it. By human evaluation and ROUGE score, their results are more consistent and informative. They also proved that increasing the entailment-aware training could improve the accuracy of the summary, however, exceeding certain threshold (20 batches in their experiment), the ROUGE-2 will drop.

Chapter 3 Methodology

This chapter introduces three experiments in this research. It will explain the overview conception of the experiments then describe each experiment's details step by step.

3.1 Introduction

As we described in previous chapters, we designed a novel objective – MSLM for pre-training Transformers. This chapter introduced three variants of MSLM (MSLM-noun, MSLM-verb, MSLM-rest). We will compare these three variants on downstream abstractive document summarization task and find which one has best performance in Chapter 4. MLM objective has been proved by many researchers (i.e.) has significant contribution for pre-training Transformers, thus, following their tradition we also designed an experiment concatenate MLM and MSLM (MLM-MSLM). We evaluated the results MLM-MSLM and only MLM on downstream task in chapter 4. We introduced our novel fine-tuning method – DADU – in this chapter, as we described in previous chapter, DADU will fine-tune one article from different aspect then merge the results, thus, we also introduced the merge function we implemented in this project. For integrity reasons, we also introduced pre-training Transformer's architecture, fine-tuning (SEQ2SEQ) method and MLM objective more detail in this chapter. We also introduced ROUGE score that we used to evaluate results on abstractive document summarization task.

3.2 ROUGE

ROUGE stands for Recall-Oriented Understudy for Gisting Evaluation (Lin, 2005). It includes a set of metrics used for automatic summary and machine translation evaluation. The metrics we used in this article are ROUGE-N and ROUGE-L. ROUGE-N measures the overlap of n-grams between automatic summary and referenced summary. The N of n-grams could be varied from 1 to n. Most widely used n-gram is unigram (ROUGE-1) and bi-gram (ROUGE-2). ROUGE-L computes the longest common sequence between label summary and machine generated summary. ROUGE score includes three factors, precision, recall, and f-measure. For ROUGE score, precision is the number of overlapping words in generated and reference summary divided by the total number of words in generated summary. In contrast, recall is divided by the number of words in reference summary. The equation showed in (2, (3, (4. We showed how to calculate Rouge score manually for individual generated summary in Chapter 4. And all the results in next chapter-results and analysis-is the average rouge score for all test articles.

$$precision = \frac{overlapping\_words}{total\_words\_in\_generated\_summary} \quad (2$$

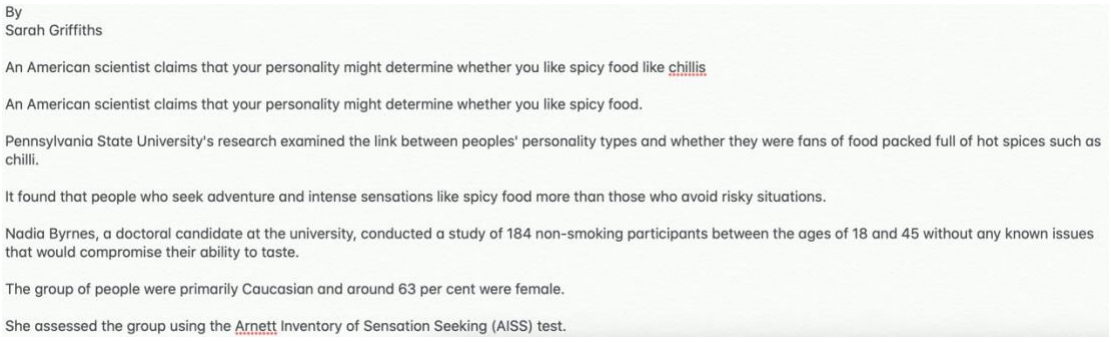
$$recall = \frac{overlapping\_words}{total\_words\_in\_reference\_summary} \quad (3$$

$$F1 = 2 * \frac{precision * recall}{precision + recall} \quad (4$$

3.3 Mask Language Model (MLM)

MLM is the main objective of Roberta pretraining. As described in the previous chapter, MLM been proposed in BERT (Devlin et al., 2018) for the first time with another objective – Next Sentence prediction (NSP). However, (Liu et al., 2019) in Roberta proved NSP did not provide significant contribution to the result, thus Roberta discarded NSP and only kept MLM.

Roberta also used the same implementation as BERT. They treated each sentence as one input, and uniformly selected 15% tokens of each sentence for replacement, randomly choose 80% of the selected tokens to replace by special symbol [MASK], 10% original, another 10% was replaced by randomly selected tokens. The target is the original words that have been replaced. MLM objective used cross-entropy loss on predicting the masked tokens. In our project, we kept our implementation of MLM same as BERT and Roberta. Example inputs shown in Figure 3:


By
Sarah Griffiths

An American scientist claims that your personality might determine whether you like spicy food like chillis

An American scientist claims that your personality might determine whether you like spicy food.

Pennsylvania State University's research examined the link between peoples' personality types and whether they were fans of food packed full of hot spices such as chilli.

It found that people who seek adventure and intense sensations like spicy food more than those who avoid risky situations.

Nadia Byrnes, a doctoral candidate at the university, conducted a study of 184 non-smoking participants between the ages of 18 and 45 without any known issues that would compromise their ability to taste.

The group of people were primarily Caucasian and around 63 per cent were female.

She assessed the group using the Arnett Inventory of Sensation Seeking (AISS) test.

Figure 3: CNN/Daily-mail - Example source text.

Then tokenize the source text by BPE. The tokens of each sentence possess one cell in the Table 1 as below.

Table 1: MLM - Example results of using BPE tokenize the source text

After tokenized by BPE
35314 5256 2543 468 3414 339 481 1302 284 307 262 1306 10106 286 8765 4345 5289 532 5149 4446 340 318 257 2863 284 705 3803 2279 329 262 1365 6

464 17695 508 1839 257 3334 3078 5373 284 4829 503 262 45571 9591 48261 286 8765 4345 5289 7415 3414 339 481 1302 355 257 705 24778 7459 29001 7635 3098 12 46260 4795 4540 287 465 1295 13
35314 5256 2543 16387 284 886 812 286 705 49 4728 48114 6 4819 13
5956 1285 1770 5256 2543 2957 1440 8850 4446 508 47437 7432 11 3176 24398 622 259 290 1743 2471 10036 284 423 406 315 38916 46611 38478 515 422 2607 416 281 13901 2184 13
…….

As in Table 2, after tokenization, MLM mask 15% tokens of each token sequence using special symbol [MASK] randomly. We add an extra item in token dictionary [MASK] and set its value equal to 25652. Each masked token sequence is an input, the output is the original token that has been masked and all the other tokens in output change to 1. Example inputs and outputs shown in below Table 2.

Table 2: MLM - Example inputs and outputs

Inputs	Outputs
35314 5256 25652 468 3414 339 481 1302 284 307 262 1306 10106 286 8765 4345 5289 532 5149 25652 340 318 257 2863 284 705 3803 2279 329 262 25652 6	1 1 2543 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 4446 1 1 1 1 1 1 1 1 1 1 1365 1
464 17695 508 25652 257 3334 3078 5373 284 4829 503 262 25652 9591 48261 286 25652 4345 5289 7415 3414 339 25652 1302 355 257 25652 24778 7459 29001 25652 3098 12 46260 4795 4540 287 465 1295 13	1 1 1 1839 1 1 1 1 1 1 1 1 45571 1 1 1 8765 1 1 1 1 1 481 1 1 1 705 1 1 1 7635 3098 1 1 1 1 1 1 1 1
35314 5256 2543 25652 284 886 812 286 705 49 25652 48114 6 4819 13	1 1 1 16387 1 1 1 1 1 1 4728 1 1 1 1
5956 1285 1770 5256 25652 2957 1440 8850 4446 508 25652 7432 11 3176 24398 622 259 290 25652 2471 10036 284 423 25652 315 38916 46611 38478 515 422 2607 416 281 25652 2184 13	1 1 1 1 2543 1 1 1 1 1 47437 1 1 1 1 622 1 1 1743 1 1 1 1 406 1 1 1 1 1 1 1 1 1 13901 1 1
…….	…….

3.3.1 Pseudocode

Set special symbol [Mask] = 25652 in dictonary file.

For loop all articles in CNN/Daily-mail dataset.

    Find out the index of '@highlight' symbol in each article.
    Remove all text after '@highlight'. # The text after '@highlight' is summary.

    Sentence list = split articles into sentence list.
    For loop sentence list
        Use BPE tokenize each sentence.

Src = Use special symbol [MASK] replace 15% tokens of each sentence randomly.

Target = Keep those tokens as original that been masked in last step and use 1 replace all the other tokens.

Input src and target into Transformer model that implemented by Fairseq

Keep training 10 epochs.

After 10 epochs, we obtained the Roberta-cnn-10epochs in this article.

3.4 Mask Summary Language Model - MSLM

The objective MSLM that we proposed is different with MLM, MLM does not need summary text, all its mask work is applied on source text. MSLM is designed for document summary tasks especially, it works on supervised dataset. MSLM aligned source text and summary text, then kept source text as original and masked certain words in summary text. Then join the source text and the preprocessed summary text as training input. The original words that have been masked in summary is the target. We designed three variants of MSLM, MSLM-noun keep noun as original and mask all other words; MSLM-verb keep verb as original; MSLM-rest mask all noun and verb, keep all other words as original. Example input and target of MSLM-noun shown in Table 3, in Table 3 red words are the summary part.

Table 3: MSLM - Example inputs and outputs

Input	Target
Singer-songwriter David Crosby hit a jogger with his car Sunday evening, a spokesman said The accident happened in Santa Ynez, California, near where Crosby lives Crosby was driving at approximately 50 mph when he struck the jogger, according to California Highway Patrol Spokesman Don Clotworthy The posted speed limit was 55 The jogger suffered multiple fractures, and was airlifted to a hospital in Santa Barbara, Clotworthy said His injuries are not believed to be life threatening [mask] happens in [mask] [mask], [mask], near where [mask] lives . The [mask] suffered multiple [mask]; his [mask] are not believed to be life-threatening .	Accident [mask] [mask] Santa Ynez, California, [mask] [mask] Crosby [mask] . [mask] jogger [mask] [mask] fractures; [mask] injuries [mask] [mask] [mask] [mask] [mask] [mask].

Then we also tokenized the input and target by BPE.

3.4.1 Pseudocode

For loop all articles in CNN/Daily-mail dataset

    Find out the index of '@highlight' symbol in each article

    source, summary = Split each article to source text and summary text by the index

    Find out all nouns in summary text.

    summary_mask = Use [Mask] replace all nouns in summary.

    summary_nouns = Use special symbol '+' replace all the other words except nouns in summary.

    source = source + summary_mask

    src = BPE_tokenize(source)

    target = BPE_tokenize(summary_nouns)

Binarization and merge all tokenized src files into one src.bin file.

Binarization and merge all tokenized target files into one target.bin file.

#Input the src.bin and target.bin into Transformer model that implemented by Fairseq.

Keep training 10 epochs.

After 10 epochs, we obtained the BQ-partial-noun in this thesis.

Same process for BQ-partial-verb and BQ-partial-rest.

3.5 MLM+MSLM

Inspired by BERT that concatenated MLM objective and NSP during training, and Roberta and BART already proved MLM provided the main contribution for state-of-art text representation, we concatenated MLM objective and our MSLM objective to train our model BQ. As described above, the input and target of MLM and MSLM is different. MLM could randomly mask words during training, however, MSLM needs to decide which kind of words to mask (i.e. only mask noun in summary), the mask work needs to be proceeded before tokenization. We proceed with MSLM training

after MLM training for one article in each epoch. Figure 4 shown the process.

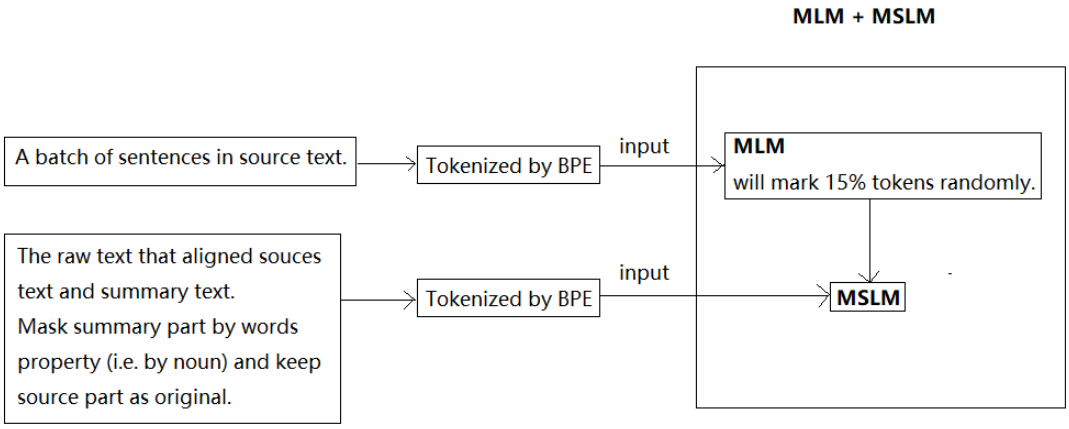


Figure 4: Concatenate MLM and MSLM

3.6 DADU

DADU design is based on traditional fine-tuning (SEQ2SEQ) method. Using traditional fine-tuning to solve downstream abstractive document summarization task, the input is the source text, the target is the summary text. Different from traditional fine-tuning, the target of DADU is not the original summary text. The target of DADU only keeps certain the words in original summary, using special symbol to mask other words. For example, in our experiment we designed DADU-noun, which meant we only kept nouns as original in the summary, using '-' instead all verbs, using '=' instead all the other words.

The generated summary after fine-tuning on this kind of summary could be viewed as one aspect (noun) of text. We designed two other variants of DADU in our experiment (DADU-verb and DADU-rest). We could generate three different version summaries for the same article. Each summary represents one view. Then using the merge function that we introduced in the next section, we merged the three different view summaries into one summary.

To easier explain the value of such design, we borrowed one example from medical science. People separate the body into different systems when researching human body, such as the skeleton system, blood circulation system, muscle and so on. Each of these systems is researched separately, then researchers merge the results from all systems to obtain the panorama of the human body and every researching progress of one system lets researchers understand the whole human body deeper. Language teacher will also let students focus on diverse types of words in different practices when teaching students reading, sometimes focusing on nouns, while sometimes the focus will be on verbs. From such focusing practice, we assume one student could obtain many views of architecture of one article subconsciously when reading an

article, then his/her brain remerges with those views into one view to understand entire article. Table 4 shown an example of source text and original summary for DADU training. Table 5 shown the targets of three sub-tasks of DADU.

Table 4: DADU - Example source and summary text

Source Text	Original Summary
Hillary Clinton has taken a jab at Vice President Joe Biden by questioning his support for the Osama bin Laden raid . The dig marks an important line of offense for Clinton as Biden would be her strongest competition if they both decide to run as the next Democratic nominee for president in 2016 . At a conference in Atlanta on Tuesday , the former Secretary of State told a version of events where she was in favor of giving the Navy SEAL mission the go - ahead to kill the terrorist but Biden was more uncertain . ……	The former Secretary of State spoke out about the decision to order the Navy SEALs to kill Osama bin Laden. Said that she pushed for it and Biden was more cautious. Both she and Biden are seen as front runners in 2016 race.

Table 5: DADU - Example targets of three sub-tasks

DADU-noun target summary: Keep all nouns as original, use '-' instead all verbs, use '=' instead all the others.	= = Secretary = State - = = = decision = - = Navy SEALs = - Osama bin Laden. - = she - = it = Biden = = =. = she = Biden = - = = runners = = race.
DADU-verb target summary: Keep all verbs as original, use '+' instead all noun, use '=' instead all the others.	= = + = + spoke = = = + = order = + + = kill + + +. Said = + pushed = + = + = = =. = + = + = seen = = + = = +.
DADU-rest target summary: Keep all other words as original, use '-' instead all verbs, use '+' instead all noun.	The former + of + - out about the + to - the + + to - + + +. - that + - for + and + was more cautious. Both + and + are - as front + in 2016 +.

Figure 5 shown the process that DADU generated three version summaries for one article, then use merge function merger three summaries into one final summary.

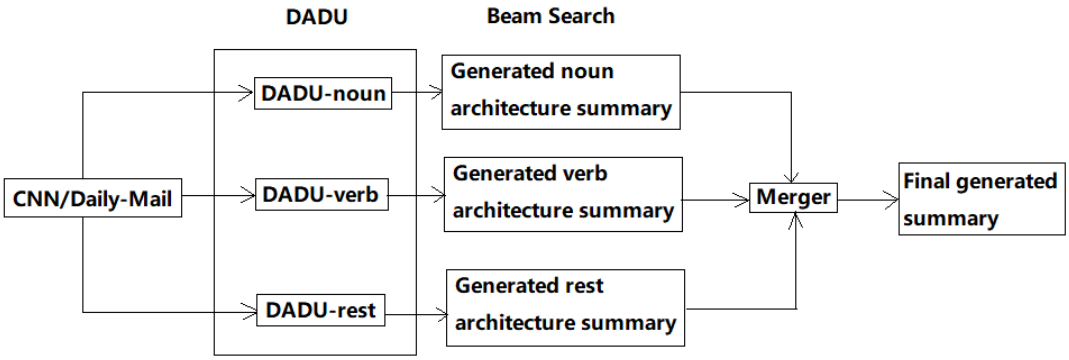


Figure 5: The process of DADU.

We trained three sub model of DADU on CNN/Daily-mail dataset. Every sub task is a seq2seq model. The we use beam search to generate different version of summaries for same article based on the three trained models. Then we use the merge function to obtain the final summary.

3.6.1 Pseudocode

```
# Create a seq2seq model, load Roberta-base model as encoder and decoder
seq2seq = create_seq2seq (Roberta, Roberta)

source_dataset = Keep all source file as original
target_dataset = Use special symbol '+' replace all nouns in summary.

# keep train seq2seq model 10 epochs
model = seq2seq(source_dataset, target_dataset)

After 10 epochs, we obtain DADU-noun model, then we use beam search to generate summaries on this model

# use beam search to generate the summary
input = tokenize(source_text)
outputs = beam_serach(model, inputs)
summary = tokenize.batch_decoder(outputs)

Same process for DADU-verb and DADU-rest
```

3.7 Merge Function

As we described in the last section, we could generate three different summaries for one article. Example generated summaries for three sub-tasks of DADU shown in Table 6:

Table 6: DADU - Example generated summaries of three sub-tasks.

Name	Summary	Word list
DADU-noun	= page - = show Transcript. - = Transcript = - students = - comprehension = vocabulary. = = bottom = = page, comment = = chance = = - = CNN Student News.	
DADU-verb	= + includes = + +. Use = + = help + = reading + = +.	[includes, use, help Reading]
DADU-rest	This + - the + +. - the + to - + with - + and +, + + + -. The +'s + - +'+ of + in the +.	[This, the, the, to, with, and, The, 's, of, in, the]

In Table 6, all the words in DADU-verb, and DADU-rest have been collected in a list by sequence and shown in word list column. The merge function merging those words into DADU-noun summary by their corresponding symbol in sequence. For example, the first symbol in DADU-noun is "=", that indict this position should be a word that been classified in 'rest'. The first word in DADU-rest word list is 'This', thus, merge function use 'This' instead '='. Example process of merge words in DADU-rest summary into DADU-noun summary shown in Figure 6 below:

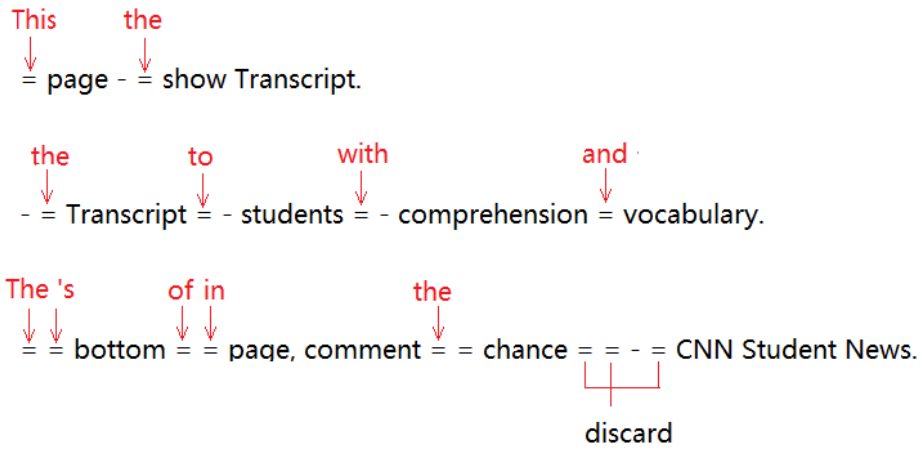


Figure 6: Merge rest into DADU-noun version summary.

In Figure 6, all "=" symbol will be replaced by word in DADU-rest word list in order. There has 14 "=" symbols in DADU-noun summary, however only 11 words in DADU-rest word list, thus the last three "=" symbol in DADU-noun summary been discard. Then merge words in DADU-verb word list into summary, the process shown in Figure 7 below:

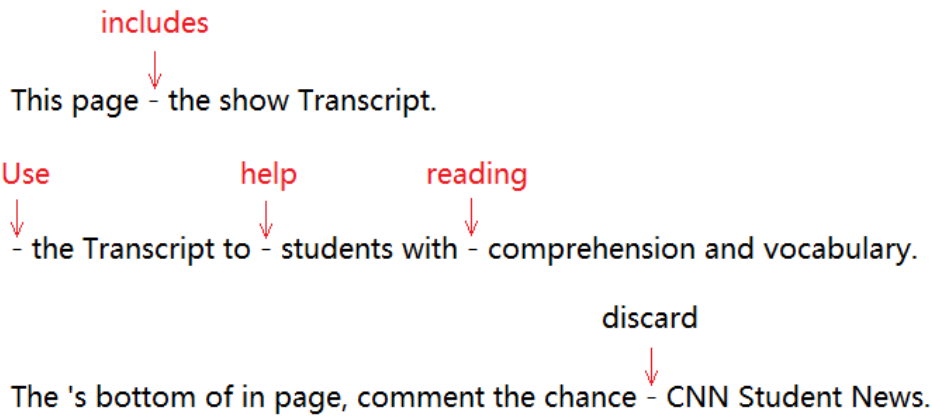


Figure 7: Merge verbs into final summary.

In Figure 7 all '-' symbols we be replaced by verb in DADU-verb word list in order. Since no word match the last '-' symbol, it been discarded.

Table 7 shown some examples of generated summaries and their corresponding rouge score by DADU. More results shown in Appendix 1.

Table 7: DADU - Example generated summary after merge.

Label: Accident happens in Santa Ynez, California, near where Crosby lives . The jogger suffered multiple fractures; his injuries are not believed to be life-threatening. Generated: a accident driving with Santa Ynez, California, his The Crosby struck multiple jogger. and jogger suffered was fractures, to airlifted a in hospital Santa Barbara.			
	Precision	Recall	F1
Rouge-1	27.88	31.53	29.39
Rouge-2	12.87	15.27	13.87
Rouge-L	27.88	31.53	29.39
Label: Sigma Alpha Epsilon is being tossed out by the University of Oklahoma . It's also run afoul of officials at Yale, Stanford and Johns Hopkins in recent months . Generated: Sigma Alpha Epsilon is under fire for a video suspended party says done fraternity members change a racist chant. It 's The of the time SAE with banned controversy the is months. permanently has to, hard alumni to chapters recently.			
	Precision	Recall	F1
Rouge-1	18.91	26.92	22.22

Rouge-2	7.69	11.53	9.23
Rouge-L	16.21	23.07	19.04
Label: Join Meerkat founder Ben Rubin for a live chat at 2 p.m. ET Wednesday . Follow @benrbn and @lauriesegallcnn on Meerkat . Use hashtag #CNNInstantStartups to join the conversation on Twitter . Generated: Meerkat launched Austin The storm on week. or at chance 2 ask on Meerkat is Twitter. his questions comments meerkat.			
	Precision	Recall	F1
Rouge-1	27.77	17.85	21.73
Rouge-2	5.26	3.44	4.16
Rouge-L	27.77	17.85	21.73
Label: Kremlin releases images of a meeting it says President Vladimir Putin held Friday . Putin spokesman dismisses rumors of ill health sparked by Putin's cancellation of planned talks . Generated: Russian photos appears Putin meeting in a head of by Supreme Court. the spokesman says with says rumors the of health. Putin the says in on Kyrgyz counterpart month.			
	Precision	Recall	F1
Rouge-1	40.9	37.5	39.13
Rouge-2	0.0	0.0	0.0
Rouge-L	31.81	29.16	30.43

3.8 Theoretical Discussion

The pre-training experiments described in this chapter such as: MLM, MSLM, MLM+MSLM will generate different models. These models could be load into Seq2Seq model. Then we fine tuning on this Seq2Seq model for 10 epochs. After 10 epochs fine-tuning, we use beam search to generate the abstractive summary for articles. We use pseudocode to clarify the process. For example, after pre-training on MLM+MSLM objective, we obtain BQ model.

BQ = pre-training (MLM+MSLM, 10 epochs)

Then we can create a Seq2Seq model, and set its encoder and decoder both BQ:

seq2seq = create_seq2seq(BQ, BQ)

Next, we fine tuning on this Seq2Seq model 10 epochs:

model = fine-tunining(seq2seq, 10 epochs)

At the end, we can use beam search to generate summary based on this Model

summary = beam_search(model, source_text)

Chapter 4 Results and Analysis

In this chapter the author of this research displayed ROUGE score for three experiments introduced in Chapter 3. The author describes the dataset and data pre-processing used in the experiments first, then describes the implementation and environment of the experiments. At the end, the author compares and evaluates the results of the three experiments.

4.1 Introduction

For abstractive document summary, the widely used evaluation method is ROUGE. In this article we choose to compare ROUGE-1, ROUGE-2, and ROUGE-L to evaluate our results. Following the tradition of other researchers, we compared F1 score for each ROUGE method, but we still showed precision and recall score for integrity reason. All the results shown in this chapter is the average rouge score of all test articles.

This chapter includes four sections, we introduced the two datasets used in our project, and described our pre-processing procedure first. Then, we showed the ROUGE score for our four experiments, evaluated and analyzed the results for each experiment. At the end, the limitation of our work is discussed

4.2 Data Description

This section will introduce the two datasets we used in our project. Amazon Food Reviews dataset and CNN/daily-mail dataset.

4.2.1 Amazon Food Reviews

This dataset consists of reviews of foods from Amazon. The data spanned a period of more than 10 years, including all ~500,000 reviews up to October 2012. Reviews include product and user information, ratings, and a plain text review. It also included reviews from all other Amazon categories. All the data was stored in a csv file – Reviews.csv. The data included reviews from Oct 1999 - Oct 2012, and 568,454 reviews in total. Each sample included the following columns shown in Table 8:

Table 8: Food Review - columns

ID	ProductId	UserId	ProfileName	Helpfulness Numerator	Helpfulness Denominator	Score	Time	Summary	Text

In our work, we focus on Summary and Text column. Example data of Summary and Text shown in Table 9:

Table 9: Food Review - Example source and summary text

Summary	Text
Good Quality Dog Food	I have bought several of the Vitality canned dog food products and have found them all to be of good quality. The product looks more like a stew than a processed meat and it smells

	better. My Labrador is finicky and she appreciates this product better than most.
Not as Advertised	Product arrived labeled as Jumbo Salted Peanuts...the peanuts were actually small sized unsalted. Not sure if this was an error or if the vendor intended to represent the product as "Jumbo".

4.2.2 CNN/Daily-Mail

This dataset is mainly for document summarization tasks. It contains 93k CNN articles and 220k Daily Mail stories, and their human generated abstractive summary as bullets. In all, the corpus has 286,817 training pairs, 13,368 validation pairs and 11,487 test pairs, as defined by their scripts. Example data shown in Table 10:

Table 10: CNN/Daily-mail - Example source and summary text

Source Text
West Ham chairman David Gold is 'hopeful' Winston Reid will snub Tottenham Hotspur and Arsenal by signing a new contract The New Zealand central defender is out of contract this summer and had looked set for a move to Tottenham on a £60,000-a-week deal while Arsenal also showed an interest Talks between West Ham and Reid have led to a contract offer and, with the club sitting ninth in the Barclays Premier League, the 26-year-old is close to agreeing to stay Winston Reid looks set to sign a new deal at West Ham after being linked with a move away in the summer Reid was linked with a move at the end of the season to either Tottenham Hotspur or Arsenal Reid has been at West Ham since 2010 after joining from Danish club FC Midtjylland In reply to a question on Twitter about whether Reid will stay, Gold replied: 'I am hopeful that Winston will sign for us ' It would reflect another step forward for West Ham as they look to qualify for Europe this season, despite losing 3-1 against Crystal Palace at home on Saturday Sam Allardyce's team welcome Chelsea to Upton Park on Wednesday night where defender Reid is likely to have his hands full against the Premier League's leaders Reid joined West Ham from Danish club FC Midtjylland in 2010 and has been an impressive performer at the heart of their defence since his arrival, playing a key role this season

West Ham chairman David Gold (right) is hopeful Reid will sign a new contract with the Premier League club West Ham chairman Gold replied to a question on Twitter about whether Reid will stay at Upton Park
Summary
Winston Reid looks set to sign a contract extension at West Ham. West Ham chairman David Gold: 'I am hopeful that Winston will sign for us' Reid is out of contract this summer and had been linked with other clubs. Tottenham Hotspur and Arsenal were among those interested. The New Zealand defender has been at Upton Park since 2010.

4.3 Data preprocessing

Since our benchmark is roberta-base and roberta-base only suits texts not longer than 514 words, we filtered out those texts longer 500 words on both datasets in this article.

4.3.1 Amazon Find Food Reviews

This dataset is only for downstream abstractive document summarization task. And for this task we only applied a traditional fine-tune method on it. Except for filtering out texts longer than 500 words, we did not apply other preprocessing on this dataset. We split the dataset to training, validation, and test pairs. Training pair included 400000 reviews, validation pair included 50000 reviews, and test pair included 50000 reviews.

4.3.2 CNN/Daily-Mail

After filtering out texts longer than 500 words, the training pair has 93383 records, validation pair has 4817 records, and the test pair had 4817 records. This dataset will be used on both pretraining Transformer - fine-tuning architecture and DADU fine-tuning method.

For pretraining Transformers - fine-tuning architecture we split it into two equal amounts, one for pretraining, another one for fine-tuning, each part contains almost the same number of articles as shown in Table 12. For the pretraining Transformers part, as we described in the methodology chapter, we designed three variants SM objective in our project, different from LM objective (LM mask words randomly, thus LM could mask words after tokenize and during training), SM needs to consider part of speech, thus we applied mask words work before tokenize, for none-SM we used [MASK] to replace all words except none words in summary, for verb-SM we used [MASK] replace all words except verb in summary, for rest-SM we used [MASK]

replace all none and verb. After the mask work, we applied GPT-2 BPE tokenize on all articles and summaries. Then we binarized all tokenized articles into src.bin file, and all tokenized summaries into three version bin files: target-noun.bin, target-verb.bin, and target-rest.bin. Binarized file is faster for memory loading operation. The fine-tuning part does not need extra preprocessing.

For DADU, as we described in the methodology chapter, we split the traditional fine-tuning into three sub tasks. The target of each task was to keep certain kind of word, and use special symbols to replace other words in summary. In this project, we used '+' replace noun, '-' replace verb, '=' replace all the other kind of words. Example original summary and preprocessed targets for three sub-tasks shown in Table 11:

Table 11: DADU - Example original summary and preprocessed targes for three sub-tasks.

Original:
Rum was made in Barbados in 1780 and shipped back to Britain. Aristocrat Henry Lascelles put 226 bottles in Harewood House , Leeds. But only a few bottles a year were drunk and then they were forgotten. Dust - covered bottles were discovered in 2011 during a stock check. Sale means each bottle sold for more than £8 , 000 making it also the most expensive rum in the world , according to an expert.
DADU-noun
Rum = - = Barbados = = = - = = Britain. Aristocrat Henry Lascelles - = bottles = Harewood House, Leeds. = = = = bottles = year = = = = they = -. Dust - - bottles = - = = = = = stock check. Sale - = bottle - = = = = =, = - it = = = = rum = = world, - = = expert.
DADU-verb
+ = made = + = = = shipped = = +. + + + put = + = + +, +. = = = = + = + = = = = + = forgotten. + - covered + = discovered = = = = + +. + means = + sold = = = = =, = making + = = = = + = = +, according = = +.
DADU-rest
+ was - in + in 1780 and - back to +. + + + - 226 + in + +, +. But only a few + a + were drunk and then + were -. + - - + were - in 2011 during a + +. + - each + - for more than £ 8, 000 - + also the most expensive + in the +, - to an +.

The statistics of these two datasets after been preprocessed shown in Table 12:

Table 12: Statistics of datasets

	Train	Val	Test
Food Revew	400,000	50000	50,000
CNN/daily-mail	93383	4817	4817
CNN/daily-mail pre-train	47383	2417	2417
CNN/daily-mail fine-tune	46000	2400	2400

4.4 Results and Analysis

All our experiments were trained on Google Colab, however, the computing resource provided by Colab is limited and unstable, thus, we restricted all our pre-training tasks to 10 epochs, and fine-tuning tasks to 1 epoch for each experiment.

A further limitation of Google Colab which led us to decide to restrict our training time is described at the limitation of the experiments section.

Compared to other researchers', our training time is much shorter (i.e. Roberta was trained over 40 epochs), the final value of our experiments is also much lower than state-of-the-art results. The ability of our experiments has not been fully explored in this article. For the purpose of proving our thinking, we re-pretrained Roberta and restricted its training time to 10 epochs. We re-pretrained Roberta on the same CNN/Daily-mail dataset that preprocessed in this article, the original Roberta was trained on Wikipedia dataset. We named the Roberta that we pretrained as Roberta-cnn-10epochs. Roberta-cnn-10epochs is a benchmark in this article. (Zou et al., 2020) released a set of abstractive summarization ROUGE scores trained on Roberta-base, Roberta-large and Bert respectively shown in Table 13. We also showed their score in each of our experiments results for easy comparation.

Table 13: Benchmark results - Bert and Roberta

	Rouge-1	Rouge-2	Rouge-L
Bert	42.13	19.60	39.18
Roberta-base	42.30	19.29	39.54
Roberta-large	43.06	19.70	40.16

Since the conception objective and model co-existed, we thought that might cause some confusion. We clarified the relationship between objective and model again in Table 14.

Table 14: Match models and objectives

Model	Objective	Dataset	Trained Epochs
Bert	MLM + NSP	Wikipedia	Approximately 40 epochs (Devlin et al., 2018)

Roberta	MLM	Wikipedia	Over 40 epochs (Liu et al., 2019)
Roberta-cnn-10epochs	MLM	CNN/Daily-mail	10 epochs
BQ	MLM + MSLM_noun	CNN/Daily-mail	10 epochs
BQ-partial-noun	MSLM_noun	CNN/Daily-mail	10 epochs
BQ-partial-verb	MSLM_verb	CNN/Daily-mail	10 epochs
BQ-partial-rest	MSLM_rest	CNN/Daily-mail	10 epochs

Table 14: Bert model was trained on MLM objective concatenate NSP objective. Roberta was trained on MLM objective only. Our model BQ was trained on MLM objective concatenate with MSLM_noun. BQ-partial-noun was trained on MSLM_noun objective only, same naming rule applied on BQ-partial-verb and BQ-partial-rest.

In the rest of this section, we use model names to compare the results.

4.4.1 Find the best among three BQ-partial variants

As we described before, we first pre-trained each BQ-partial variant on CNN/Daily-mail pre-train dataset for 10 epochs. Then we fine-tuned them on FoodReview and CNN/Daily-mail fine-tune dataset one epoch, respectively. The results shown in Table 15.

Table 15: Results – Three BQ-Partial variants

		BQ-partial-noun	BQ-partial-verb	BQ-partial-rest
FoodReview	Rouge-1	18.22	15.17	14.79
	Rouge -2	5.28	4.11	3.83
	Rouge -L	18.02	14.99	14.65
CNN/Daily-mail-fine-tune	Rouge -1	24.95	5.64	3.27
	Rouge -2	0.77	0.21	0.12
	Rouge -L	23.63	5.31	3.07

Table 15: For both FoodReview and CNN/Daily-mail-fine-tune, BQ-partial-noun obtain the best F1 score for all ROUGE method. Since BQ-partial-noun was trained on MSLM_noun objective, we choose concatenate MLM and MSLM_noun to train our final BQ model.

The reason that BQ-partial-noun obtained the best score might be because noun words usually indicate the entities of the article. The verb and all the other words such as: conjunction and adjective usually indicate the relationship between entities. Our experiments could partially prove noun words are more important features of articles.

4.4.2 Roberta-cnn-10epochs vs BQ

As we described above BQ was trained on MLM concatenate MSLM_noun, Roberta-cnn-10epochs was trained on MLM only. We pre-trained both of these two models on CNN/Daily-mail pre-train dataset for 10 epochs. Then we fine-tune them on CNN/Daily-mail fine-tune dataset for one epoch. The results shown in Table 16.

Table 16: Results - BQ and Roberta-cnn-10epochs

Model	Rouge-1	Rouge-2	Rouge-L
Roberta-cnn-10epochs	27.53	8.41	19.68
BQ	30.18	9.67	21.35
Bert	42.13	19.60	39.18
Roberta-base	42.30	19.29	39.54
Roberta-large	43.06	19.70	40.16

Table 16: Compared to Roberta-cnn-10epochs, BQ improves 2.65% for ROUGE-1, 1.26% for ROUGE-2, and 1.67% for ROUGE-L.

One interesting part in Table 16 is the improvement of BQ. That means for abstractive summarization task, MSLM_noun objective has contributed to the results. On the other side, the score of Bert is lower than Roberta. As we described before, Bert was trained on MLM and NSP, Roberta prove NSP does not have much contribution, so they removed it and trained on MLM only. We proved our MSLM_noun has contribution, that gives us confidence if we keep training, we might obtain new state-of-art results.

The reason that MSLM_noun did contribute could be because our model has a stronger ability to conclude the kernel noun words (entities) from source text. And, letting the model focus on learning certain part of knowledge might be easier than learning knowledge as a whole, such as from previous sentences to predict the next sentence (NSP).

However, Bert and Roberta were trained on Wikipedia, then fine-tuned on CNN/Daily-mail. Our Roberta-cnn-10epochs and BQ were trained on CNN/Daily-mail pre-trained dataset, then fine-tuned on CNN/Daily-mail fine tune dataset (We split the CNN/Daily-mail to pre-train and fine-tune part, more explanation is described in the data preprocess section). That might cause the results to bias.

4.4.3 DADU vs Traditional Fine-tuning

As described in chapter 3 DADU has three sub-models. In our experiment, each sub-task was fine-tuned based on Roberta-base model with CNN/Daily-mail dataset for five epochs. Then we used our merge function to merge the summaries that generated

from these three fine-tuned sub-models. For comparison reasons, we also fine-tuned a traditional seq2seq model based on Roberta-base with CNN/Daily-mail dataset for 5 epochs. Results shown in Table 17:

Table 17: Results - DADU

		Precision	Recall	F-measure
Traditional	Rouge-1	33.15	42.46	36.02
	Rouge -2	15.95	18.97	16.58
	Rouge -L	24.89	31.44	26.77
DADU	Rouge -1	**37.10**	28.22	31.10
	Rouge -2	8.5	6.38	7.04
	Rouge -L	**25.20**	19.31	21.20

Table 17: DADU got almost 4% improvement on precision value of Rouge-1, and 0.3% improvement on Rouge -L. However, DADU got much lower score of Rouge-2 and lower score for all recall value, that make the F1 score lower than traditional fine-tuning method.

As we described in methodology, precision indicates the proportion of sequence that exists in machine generated summary, also shown in reference summary. Recall value indicates the opposite meaning. For Rouge-1 our precision value is much higher than the traditional method. And since all our precision value is higher than the recall value, traditional method reverses. One explanation is DADU tends to generate a shorter summary than reference summary, traditional tends to generate longer. Because, as we described in Chapter 3, rouge score counts the overlapping word rates between generated summary and reference summary. Precision is the number of overlapping words divided by number of words in generated summary, the recall is the number of overlapping words divided by the number of words in reference summary. If precision is higher than recall, this means the generated summary is shorter, if recall is higher than precision, it means reference summary is shorter.

We counted the rouge score manually on some examples of generated summaries that were generated by DADU in next section.

Our Rouge-2 value is much lower than the traditional method. As described in the methodology chapter, Rouge-2 means two words (bi-gram) overlapping. One plausible reason is since our merge function is rough, the generated summary might have the same amount of overlapped single words as the traditional method, but the order is wrong. Our Rouge-1 and Rouge-L value does not drop as Rouge-2 could support our reason.

For training time, since we have three sub-models, our training time is three times longer than the traditional method. However, the most state-of-art value (43.06) on traditional method was obtained by 30 epochs fine-tuning. After 30 epochs keep

training does not provide contribution. DADU might provide a method to keep increasing Rouge value by long time (i.e. 90 epochs) training, since at least we could increase precision value.

4.4.4 Example Rouge score process on generated summary

In this section, we count rouge scores manually on some of generated summaries. All the generated summaries used in this section come from our third experiment DADU and could be found in Appendix 2.

For Rouge-1, in the Figure 8, there are 9 overlapping words that we mark as orange in our generated and reference summary. The total word in generated summary is 24, in reference summary is 27. The Rouge-1 score for this generated summary is:

$$precision = \frac{9}{24} = 37.5\% \qquad recall = \frac{9}{27} = 33.33\%$$

$$F1 = 2 * \frac{precision * recall}{precision + recall} = 2 * \frac{0.375 * 0.3333}{0.375 + 0.3333} = 35.29\%$$

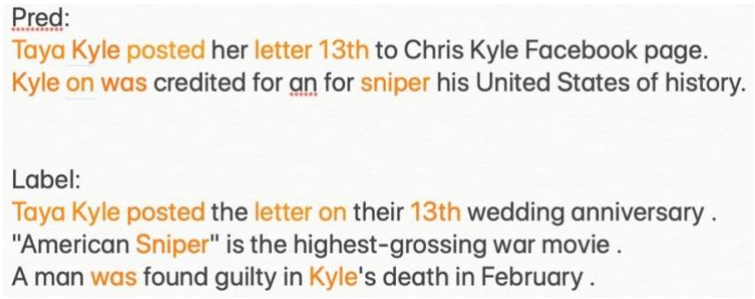


Figure 8: DADU – Example Rouge-1 for generated and reference summary

For Rouge-2, in Figure 9, rouge change the generated and reference summary to bigrams first. There are 2 common bigrams that we mark as orange. The total bigrams in generated summary is 22, in reference summary is 24. The Rouge-2 score for this generated summary is:

$$precision = \frac{2}{22} = 9.09\% \qquad recall = \frac{2}{24} = 8.33\%$$

$$F1 = 2 * \frac{precision * recall}{precision + recall} = 2 * \frac{0.0909 * 0.833}{0.0909 + 0.0833} = 8.69\%$$

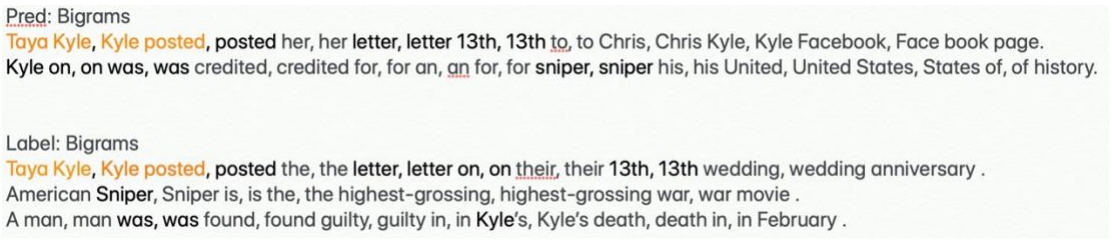


Figure 9: DADU – Example Rouge-2 for generated and reference summary

For Rouge-L, in Figure 10, the longest common sequence between our generated and reference summary is a 3 words sequence that we mark as orange. The sequence they belonged to has 11 and 10 words for generated and reference summary respectively that we mark as bold and italic. The Rouge-L score for this generated summary is:

$$precision = \frac{3}{11} = 27.27\% \qquad recall = \frac{3}{10} = 30\%$$

$$F1 = 2 * \frac{precision * recall}{precision + recall} = 2 * \frac{0.2727 * 0.3}{0.2727 + 0.3} = 28.57\%$$

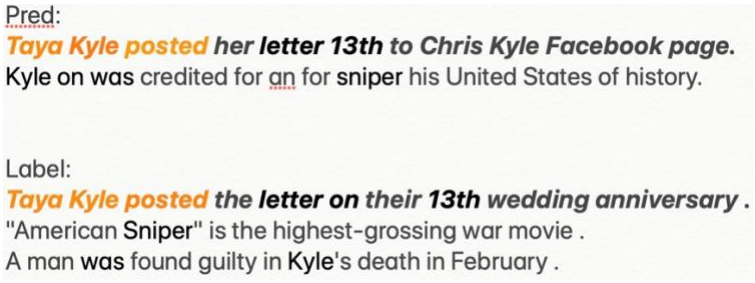
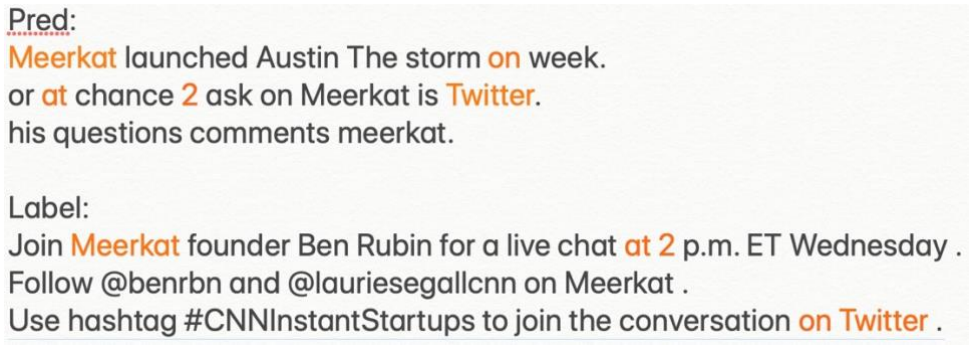


Figure 10: DADU – Example Rouge-L for generated and reference summary

Above generated summary obtains medium Rouge score for DADU. We also list another two generated summaries in Table 18 that has lower and higher Rouge score for DADU.

Table 18: DADU - Example generated summary obtain lower and higher Rouge score

Lower



	Precision	Recall	F1
Rouge-1	5 / 20 = 25%	5 / 29 = 17.24%	20.4%
Rouge-2	0	0	0
Rouge-L	1 / 7 = 14.28%	1 / 14 = 7.14%	9.25%

Higher

	Precision	Recall	F1
Rouge-1	10 / 25 = 40%	10 / 23 = 43.47%	41.66%
Rouge-2	2 / 23 = 8.69%	2 / 21 = 9.52%	9.08%
Rouge-L	2 / 6 = 33.33%	2 / 5 = 40%	36.36%

4.5 Limitation of the Experiments

The first main limitation of the experiments in this project is computing resources limits. As we described in experiments chapter, for our project the better plan was to compare our model with original Roberta, however restricted by computing resources, we re-pre-training Roberta with CNN/Daily-mail dataset for 10 epochs same as we pre-training our model BQ. Such design could partially evaluate the performance of BQ, but not fully.

The second limitation in the project is the merge function of DADU. The current merge function is too simple, we just used three special symbols to indicate the position of corresponding words when merging. We do not add any linguistic and grammar design into it.
For fully evaluating the performance of our model, the better plan is to compare it with original Roberta, however Roberta was trained on 8 * 32 GB Nvidia V100 GPUs. We do not have enough computing resources. All our experiments were trained on Google Colab. The fastest GPU on Colab is a Tesla T4 with 15GB memory, and Colab only assigns 12 hours at most in one day for each free user to connect GPU (Google, n.d.). For our experiments, each training epoch lasted around three hours when we connected to the best GPU Tesla T4. Based on our experience, Colab usually will not assign 12 hours to free users in one day, our experiments were disconnected after 10 hours and we were notified of the connection limitation. We could not always could connect to the best Tesla T4 GPU, and it seems the more frequently we used Colab GPU the more connection limitations we had. On some days, after five hours, we received notification of connection limitation and were disconnected. We tried to buy Google Colab Professional, however, Colab pro is not available in New Zealand (Google, n.d.).

Chapter 5 Conclusion and Future Work

We designed a novel SM objective and introduced our BQ model that trained on it in this article. We designed two experiments based on SM. We also introduced a novel finetuning method DADU and designed an experiment to evaluate DADU with Tradition fine-tuning method on abstractive document summary task. In this chapter we will summarize this thesis and introduce the plan for our future work.

5.1 Conclusion

Fine-tuning on pre-trained Transformers model take downstream NLP tasks to a new state-of-art. Researchers proposed many objectives such as MLM, NSP to pre-train their own Transformer models. This thesis proposed a novel objective – MSLM, and pre-trained our Transformer model BQ up on it. We designed an experiment to compare the performance of three variants of MSLM, and we found that MSLM - noun has the best performance. Since MLM is a widely used objective and has been proved by other researchers to provide a significant contribution to the pre-trained Transformer model. We designed an experiment to compare three Transformer models that trained on MLM, MSLM-noun and MLM & MSLM-noun, respectively. We found that our model BQ that pretrained on MLM & MSLM-noun obtain higher rouge score than Roberta-cnn-10epochs that pretrained on MLM only. That proved our new objective MSLM did contribution to the final results.

Besides MSLM, we proposed a new fine-tuning method DADU in our thesis, the main difference of DADU from traditional fine-tuning method is DADU can fine-tune an article from a different view first. Each view could generate a different version of the summary. DADU uses merge function to merge all view summaries into one summary. Compared to traditional fine-tuning method, DADU obtained better value of precision of ROUGE-1 and ROUGE-L.

5.2 Future Works

Our future works include three directions:

As we described in Chapter 4 limitation section, restricted by computing resources, the experiments in this article have not been trained long enough. Thus, the performance and ability of SM and DADU has not been fully explored and evaluated. In our future work, we will invest in GPU resource, to train our model and method at least as long as Roberta training.

Compared to other researchers', our project only trained on two dataset Food Reviews and CNN/Daily-mail. In our future work, we will test our experiment on more document summary dataset. Such as: XSum (Narayan et al., 2018), NEWSROOM (Grusky et al., 2018), Multi-News (Fabbri et al., 2019).

Currently, the merge function of DADU is too simple. In our future work, we will take linguistic and grammar into account.

Reference

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Zhang, J., Zhao, Y., Saleh, M., & Liu, P. (2020, November). Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning* (pp. 11328-11339). PMLR.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Jiang, Y., Celikyilmaz, A., Smolensky, P., Soulos, P., Rao, S., Palangi, H., ... & Gao, J. (2021). Enriching Transformers with Structured Tensor-Product Representations for Abstractive Summarization. *arXiv preprint arXiv:2106.01317*.

Lin, C. (2005). Recall-oriented understudy for gisting evaluation (rouge). *Retrieved August*, *20*, 2005.

Zou, Y., Zhang, X., Lu, W., Wei, F., & Zhou, M. (2020). Pre-training for Abstractive Document Summarization by Reinstating Source Text. *arXiv preprint arXiv:2004.01853*.

Song, K., Tan, X., Qin, T., Lu, J., & Liu, T. Y. (2019). Mass: Masked sequence to sequence pre-training for language generation. *arXiv preprint arXiv:1905.02450*.

Lample, G., & Conneau, A. (2019). Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

See, A., Pappu, A., Saxena, R., Yerukola, A., & Manning, C. D. (2019). Do massively pretrained language models make better storytellers?. *arXiv preprint arXiv:1909.10705*.

Welleck, S., Kulikov, I., Roller, S., Dinan, E., Cho, K., & Weston, J. (2019). Neural text generation with unlikelihood training. *arXiv preprint arXiv:1908.04319*.

Holtzman, A., Buys, J., Du, L., Forbes, M., & Choi, Y. (2019). The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Paulus, R., Xiong, C., & Socher, R. (2017). A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*.

Gehrmann, S., Deng, Y., & Rush, A. M. (2018). Bottom-up abstractive summarization. *arXiv preprint arXiv:1808.10792*.

Rothe, S., Narayan, S., & Severyn, A. (2020). Leveraging pre-trained checkpoints for sequence generation tasks. *Transactions of the Association for Computational Linguistics*, *8*, 264-280.

Pasunuru, R., & Bansal, M. (2018). Multi-reward reinforced summarization with saliency and entailment. *arXiv preprint arXiv:1804.06451*.

Arumae, K., & Liu, F. (2019). Guiding extractive summarization with question-answering rewards. *arXiv preprint arXiv:1904.02321*.

Kryściński, W., McCann, B., Xiong, C., & Socher, R. (2019). Evaluating the factual consistency of abstractive text summarization. *arXiv preprint arXiv:1910.12840*.

Wang, A., Cho, K., & Lewis, M. (2020). Asking and answering questions to evaluate the factual consistency of summaries. *arXiv preprint arXiv:2004.04228*.

Narayan, S., Cohen, S. B., & Lapata, M. (2018). Ranking sentences for extractive summarization with reinforcement learning. *arXiv preprint arXiv:1802.08636*.

Dai, A. M., & Le, Q. V. (2015). Semi-supervised sequence learning. *Advances in neural information processing systems*, *28*, 3079-3087.

Ji, Y., Cohn, T., Kong, L., Dyer, C., & Eisenstein, J. (2015). Document context language models. *arXiv preprint arXiv:1511.03962*.

Al-Rfou, R., Choe, D., Constant, N., Guo, M., & Jones, L. (2019, July). Character-level language modeling with deeper self-attention. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 33, No. 01, pp. 3159-3166).

Mikolov, T., & Zweig, G. (2012, December). Context dependent recurrent neural network language model. In *2012 IEEE Spoken Language Technology Workshop (SLT)* (pp. 234-239). IEEE.

Fabbri, A. R., Li, I., She, T., Li, S., & Radev, D. R. (2019). Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. *arXiv preprint arXiv:1906.01749*.

Maynez, J., Narayan, S., Bohnet, B., & McDonald, R. (2020). On faithfulness and factuality in abstractive summarization. *arXiv preprint arXiv:2005.00661*.

Narayan, S., Cohen, S. B., & Lapata, M. (2018). Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., & Salakhutdinov, R. (2019). Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

Kryściński, W., Keskar, N. S., McCann, B., Xiong, C., & Socher, R. (2019). Neural text summarization: A critical evaluation. *arXiv preprint arXiv:1908.08960*.

Pasunuru, R., Guo, H., & Bansal, M. (2017, September). Towards improving abstractive summarization via entailment generation. In *Proceedings of the Workshop on New Frontiers in Summarization* (pp. 27-32).

Guo, H., Pasunuru, R., & Bansal, M. (2018). Soft layer-specific multi-task summarization with entailment and question generation. *arXiv preprint arXiv:1805.11004*.

Cao, Z., Wei, F., Li, W., & Li, S. (2018, April). Faithful to the original: Fact aware neural abstractive summarization. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 32, No. 1).

Li, H., Zhu, J., Zhang, J., & Zong, C. (2018, August). Ensure the correctness of the summary: Incorporate entailment knowledge into abstractive sentence summarization. In *Proceedings of the 27th International Conference on Computational Linguistics* (pp. 1430-1441).

Chen, Y. C., & Bansal, M. (2018). Fast abstractive summarization with reinforce-selected sentence rewriting. *arXiv preprint arXiv:1805.11080*.

Falke, T., Ribeiro, L. F., Utama, P. A., Dagan, I., & Gurevych, I. (2019, July). Ranking generated summaries by correctness: An interesting but challenging application for natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (pp. 2214-2220).

Google. (n.d.). Colaboratory: Frequently Asked Questions. Retrieved January 20, 2022, from https://research.google.com/colaboratory/faq.html

Google. (n.d.). Where are Colab Pro and Pro + available?. Retrieved January 20, 2022, from https://colab.research.google.com/signup

Appendix 1

Pretrained on MLM & MSLM, then fine-tuning on FoodReview dataset.

```
---------------------------------
Pred:Good product, good price
Label:EXCELLENT PRODUCT
---------------------------------
Pred:Not worth the price
Label:Excellent with Full-Bodied Reds
---------------------------------
Pred:Yummy
Label:BRIX chocolate
---------------------------------
Pred:Very good tea.
Label:Best Iced Tea Ever
---------------------------------
Pred:Love this tea!
Label:Best TEA!
---------------------------------
Pred:Very good tea
Label:This makes the best sweet tea!!!
---------------------------------
Pred:I love this tea
Label:The secret to good Southern Sweet Tea.
---------------------------------
Pred:Love this tea
Label:Delicious!
---------------------------------
Pred:Great tea, great price
Label:Best Iced Tea!
---------------------------------
Pred:Yummy!
Label:Perfect delivery
---------------------------------
Pred:Best tea ever
Label:great tea
---------------------------------
Pred:Great tea!
Label:Luzianne Tea
---------------------------------
Pred:Best tea ever!
Label:Finally! Decent Sweet Tea in Seattle!
```

```
---------------------------------
Pred:Best tea ever!
Label:Buy this!
---------------------------------
Pred:I love this tea!
Label:Kind of old
---------------------------------
Pred:Great tea!
Label:NOT FAMILY SIZE!!! Be Careful
---------------------------------
Pred:Love it!
Label:fast
---------------------------------
Pred:Best tea ever!
Label:Luzianne Tea is the best
---------------------------------
Pred:Love this tea!
Label:good tea
---------------------------------
Pred:Tastes great
Label:Love this tea.  Makes PERFECT ice tea.
---------------------------------
Pred:Stash Tea Bags
Label:Right tea, wrong bag
---------------------------------
Pred:Tastes good
Label:Bags do not filter well.  Tea has a bad, woody taste.
---------------------------------
Pred:Great product, great price
Label:One of childhood favorite cookies
---------------------------------
Pred:Yummy!
Label:Awesome treat
---------------------------------
Pred:yummy cookies
Label:try to eat only one
---------------------------------
Pred:Yummy!
Label:These Are Amazing Treats!
---------------------------------
Pred:Do not buy this product from Amazon
Label:$7 per box of cereal???
---------------------------------
Pred:Yummy!
```

Label:Awesomeness for the tired taste buds.

Pred:Tastes great!
Label:I LOVE this stuff!

Pred:Tastes great
Label:Good but a few notes:

Pred:My daughter loves it!
Label:Olde Thompson Tex-Mex Chipotle seasoning

Pred:Good, but not great
Label:Great Oil; Reasonable Price

Pred:Delicious!
Label:Best oil I've tasted outside of Liguria

Pred:Great Olive Oil
Label:Excellent Extra Virgin Olive Oil

Pred:Great product, great price
Label:A Very Good Oil

Pred:Delicious!
Label:Olio Carli Extra Virgin Olive Oil

Pred:Good price, good product
Label:Not "fruity" tasting

Pred:Very good product
Label:EXCELLENT BUY !

Pred:Tastes great!
Label:Bland and horrible

Pred:Good, but not as good as the cracker
Label:Good crackers...bad packaging

Pred:I like them, but these are good
Label:Rye and wheat ones are best crackers EVER

Pred:Yummy!
Label:tasty,

Pred:Not what I expected.
Label:Middling at best

Pred:Great product, great price
Label:Good as any other

Pred:Good, but not great
Label:Vanilla Pleasure

Pred:Great coffee, great price
Label:Outstanding.

Pred:Great product, great price
Label:Awesome!

Pred:Yummy!
Label:Great purchase

Pred:Very good tea!
Label:Very good - Green Tea Frappuccino addict

Pred:I like this tea
Label:Great for the price & health benefits!

Pred:Great quality tea
Label:Not the best quality, but okay for the price

Pred:Very good tea
Label:Good deal on green tea...

Pred:Good tea!
Label:Great tea!

Pred:Tastes great
Label:So long Starbucks!

Pred:Great tea, great price
Label:Good for the price!

Pred:Great tea, great price
Label:Great service, fresh tasty product

Pred:Good value, good taste
Label:Quality is just ok. Shipping was super fast!

```
---------------------------------
Pred:Dogs love them!
Label:half the price in China Town
---------------------------------
Pred:Yummy!
Label:Such good service
---------------------------------
Pred:Great service!
Label:Don't buy from these people!
---------------------------------
Pred:Love this tea
Label:Good tea
---------------------------------
Pred:YUMMY!
Label:Worth twice the price!
---------------------------------
Pred:Great price, great price
Label:Oh so good!
---------------------------------
Pred:Great Product!
Label:Awesome deal
---------------------------------
Pred:Tasty tea
Label:Price is OK, flavor a bit bitter
---------------------------------
Pred:Great tea, great price
Label:Good for the price
---------------------------------
Pred:Great Green Tea
Label:Matcha Green Tea Powder is great
---------------------------------
Pred:Love this product!
Label:Matcha is the best!
---------------------------------
Pred:good deal
Label:$45 for two - rid off big time
---------------------------------
Pred:Love this tea
Label:Tastes like mud
---------------------------------
Pred:Love this stuff!
Label:Awesome.
---------------------------------
Pred:Good but not great.
```

Label:Tastes good,dissolves poorly

Pred:Disappointing

Label:This is very LOW QUALITY matcha

Pred:Not bad but not great

Label:Not Matcha!

Pred:Good but not great

Label:Good for the price

Pred:Yummy!

Label:great for the price

Pred:great tea!

Label:i really like this tea

Pred:Good tea, good price

Label:Good Quality

Pred:Great for the price!

Label:Tastes awful but it's good for you

Pred:Wonderful Tea

Label:Exceptionally rich in Anti-Oxidants,

Pred:tea time

Label:Very Pleased

Pred:I like it.

Label:Very good for green tea latte!!

Pred:Good stuff!

Label:3.5 stars

Pred:Great Product!

Label:Matcha Tea

Pred:Tastes great

Label:Tastes good, Good for you, Reasonably Priced

Pred:Great tea!

Label:love this tea!

Pred:Great value!
Label:Great Green

Pred:Great tea, great price
Label:matcha green tea smoothies!

Pred:I like it!
Label:Love it!

Pred:Tastes good!
Label:tea

Pred:Delicious!
Label:Smooth and fresh!

Pred:Love this tea
Label:I like it! beautiful container and easy to use

Pred:Great tasting tea!
Label:Well worth purchasing this tea.

Pred:I like this tea
Label:Good For Upping the Nutritional Ante

Pred:Not what I expected
Label:Great tea for the price

Pred:Great Product!
Label:Great for the price

Pred:Love this tea!
Label:Great tea at a great price!

Pred:Not what I expected
Label:Great Product!!

Pred:Great product, great service
Label:Great product/vender

Pred:Great product, great price
Label:Love this product!

Pred:Wonderful product
Label:green tea powder- met every expectation and more

Pred:Not my cup of tea
Label:Better then Starbucks

Pred:excellent product
Label:good stuff

Pred:Love this tea!
Label:Good stuff!

Pred:Not what I expected
Label:Interesting...

Pred:The best!
Label:Most bang for the buck!

Pred:Not my favorite tea
Label:Tastes like Seaweed

Pred:My cat loves it!
Label:My cat loves this and it's actually good for her.

Pred:Yummy!
Label:Surprisingly Rich

Pred:Best Cocoa Ever
Label:comfort cocoa

Pred:Tasty, but not too sweet.
Label:Necco, what have you done?

Pred:Not what I expected
Label:One word..... bleckkkk.

Pred:Not what I expected
Label:If It Ain't Broke, Don't Fix It

Pred:Disappointing
Label:Disgusting, and a destroyed holiday tradition

Pred:Tastes great
Label:sorry NECCO, the new flavors are gross

Pred:Yummy!

Label:I have a Sweet tooth

Pred:YUMMY!

Label:Valentine's Day...all year long

Appendix 2

DADU merged summaries

Pred:

a accident driving with Santa Ynez, California, his The Crosby struck multiple jogger.

and jogger suffered was fractures, to airlifted a in hospital Santa Barbara.

Label:

Accident happens in Santa Ynez, California, near where Crosby lives .

The jogger suffered multiple fractures; his injuries are not believed to be life-threatening .

Pred:

Sigma Alpha Epsilon is under fire for a video suspended party says done fraternity members change a racist chant.

It 's The of the time SAE with banned controversy the is months.

permanently has to, hard alumni to chapters recently.

Label:

Sigma Alpha Epsilon is being tossed out by the University of Oklahoma .

It's also run afoul of officials at Yale, Stanford and Johns Hopkins in recent months .

Pred:

Meerkat launched Austin The storm on week.

or at chance 2 ask on Meerkat is Twitter.

his questions comments meerkat.

Label:

Join Meerkat founder Ben Rubin for a live chat at 2 p.m. ET Wednesday .

Follow @benrbn and @lauriesegallcnn on Meerkat .

Use hashtag #CNNInstantStartups to join the conversation on Twitter .

Pred:

in has health care worker positive Sierra Leone for tested are whether Ebola.

the was majority to the, assessing cases evacuate nations.

Label:
Spokesperson: Experts are investigating how the UK military health care worker got Ebola .
It is being decided if the military worker infected in Sierra Leone will return to England .
There have been some 24,000 reported cases and 10,000 deaths in the latest Ebola outbreak .

Pred:
This page includes the show Transcript.
Use the Transcript to help students with reading comprehension and vocabulary.
bottom page, comment chance must CNN Student News.

Label:
This page includes the show Transcript .
Use the Transcript to help students with reading comprehension and vocabulary .
At the bottom of the page, comment for a chance to be mentioned on CNN Student News. You must be a teacher or a student age 13 or older to request a mention on the CNN Student News Roll Call.

Pred:
in January are, to month their set this record have sea lions in year.
more than 500 000 pups says, NOAA rescued.
SeaWorld sea lion population.

Label
"There has been an unusually high number of sea lions stranded since January," NOAA representative says .
The speculation is mothers are having difficulty finding food, leaving pups alone too long or malnourished .

Pred:
30 abducted year held old Denise Louise Huskins was say from a is acquaintance 's residence.
Huskins'whereabouts not a, police.
Police in public 's help the Huskins'welfare.

Label:
Denise Huskins, 30, works as a physical therapist at a Kaiser Hospital, CNN affiliate KGO reports .
Huskins was taken from her boyfriend's residence, her cousin tells CNN affiliate KPIX .

Pred:

two roof have cement factory collapses been Mongla, alive port city from Dhaka.
the The around people 1 trapped in southwestern says, police owned.
Incident say place p. m. (a. m).
incidents factories buildings nothing.

Label:
More than 60 are injured, many critically, official says .
The cement factory was owned by a Bangladeshi army welfare organization .
In recent years, Bangladesh has had other deadly incidents at factories .