Improving Coverage and Novelty of Abstractive Text Summarization Using Transfer Learning and Divide and Conquer Approaches

Ayham Alomari[1], Norisma Idris[2], Aznul Qalid Md Sabri[3*], Izzat Alsmadi[4]

[1,2,3*]Department of Artificial Intelligence, Faculty of Computer Science and Information Technology, Universiti Malaya, 50603 Kuala Lumpur, Malaysia

[4] Department of Computing and Cybersecurity, Texas A&M San Antonio, 78224 San Antonio, United States

Email: wva180011@siswa.um.edu.my[1] , norisma@um.edu.my[2], aznulqalid@um.edu.my [3*] (Corresponding author), izzat.alsmadi@tamusa.edu[4]

ABSTRACT

Automatic Text Summarization (ATS) models yield outcomes with insufficient coverage of crucial details and poor degrees of novelty. The first issue resulted from the lengthy input, while the second problem resulted from the characteristics of the training dataset itself. This research employs the divide-and-conquer approach to address the first issue by breaking the lengthy input into smaller pieces to be summarized, followed by the conquest of the results in order to cover more significant details. For the second challenge, these chunks are summarized by models trained on datasets with higher novelty levels in order to produce more human-like and concise summaries with more novel words that do not appear in the input article. The results demonstrate an improvement in both coverage and novelty levels. Moreover, we defined a new metric to measure the novelty of the summary. Finally, the findings led us to conclude that the novelty levels are more significantly influenced by the training dataset itself, as in CNN/DM, than by other factors like the training model or its training objective, as in Pegasus.

Keywords: Abstractive Summarization, Novelty, Coverage, Warm-Started Models, Transfer Learning, Deep Learning

1.0 INTRODUCTION

Automatic Text Summarization (ATS) is one of the most valuable systems that benefit humanity. It is utilized in various fields, including medicine [1], business [2], and education. Current research progress in leveraging State-of-the-Art (SotA) pre-trained language models demonstrates impressive improvements in both extractive and abstractive types of ATS. However, in contrast to human-generated summaries, the abstractive ATS area still has several challenges, including coverage and novelty. Coverage refers to the amount of the source text that is covered in summary, whereas novelty refers to the proportion of the summary that is not stated verbatim in the source text.

Abstractive ATS datasets vary based on the length of the input documents. They can be categorized into short-length documents (DUC [1]and Gigaword [3], [4]), medium-length documents (CNN/DM [5], [6], XSUM [7], and Reddit_TIFU [8]), and long-length documents (arXiv and PubMed [9]). In this study, we will focus on increasing the coverage of the summary by utilizing the output design of each dataset. In addition, datasets vary according to their novelty levels. In medium-length documents, for instance, the CNN/DM dataset, which is the most widely used dataset in the abstractive ATS research field, has lower novelty degrees than XSum and Reddit TIFU due to the tendency of its reference summaries towards extractive rather than abstractive summarization [10].

Intuitively, coverage is inversely correlated to the length of the source text; hence, condensing lengthy documents into summaries may result in the omission of crucial details. In addition, the study [8] noted that models trained on news article datasets, such as CNN/DM, generate summaries that emphasize mainly the beginning of the input article. The properties of the labeled summaries of the training dataset influence that behavior. Moreover, most research work is limited to the capacity of the employed pre-trained models, typically 512 tokens, resulting in the loss of the remainder of the article's information. In this research, in order to solve the abovementioned issues, the divide and conquer strategy is utilized to partition the entire input document into *n* sections and then to summarize each portion using models trained on datasets with higher levels of novelty. As a result, summarizing each section individually leads to a focused summary of the section's specific details. Then, by combining all the generated summaries, a comprehensive summary of the input article's essential details, regardless of their position in the input article, is

[1] http://duc.nist.gov/

produced. On the other hand, applying models trained on the XSum and Reddit-TIFU corpora to the CNN/DM dataset provides more human-like summaries that are recognized for their use of novel phrases.

Consequently, the objective of this research is to improve the coverage and novelty of the outcomes tested on the CNN/DM dataset by employing the divide-and-conquer approach and models trained on higher novelty datasets. The key contributions of this study are summarized below:

1- Finetune three warm-started models on XSum and CNN/DM datasets: bert2gpt_xsum, roberta2gpt_xsum, and roberta2reoberta_cnndm.
2- The use of divide-and-conquer and transfer learning approaches to boost coverage and novelty levels, respectively.
3- Define a new novelty metric that overcomes the shortcomings of existing metrics and yields more accurate scores.

2.0 LITERATURE REVIEW

2.1 SotA Abstractive ATS Models

In recent years, abstractive ATS task has gained considerable research attention. First, topic-based [11], statistical-based [12], graph-based [13], and discourse-based [14] approaches have begun to generate summaries automatically. Then, machine learning approaches, including supervised [15], and unsupervised [16], arose in the domain. After the availability of massive datasets and GPUs, deep learning approaches have recently started to be implemented. As a result, there was a significant enhancement, particularly with the collaboration with reinforcement learning approaches, which have been used to enhance the results and solve various problems, such as exposure bias [17], loss/evaluation mismatch [17], and novelty problems [18], [19]. The results were not comparable to human-generated summaries until the emergence of Transformers [20] and pre-trained language models [21]–[23] utilizing Transfer Learning (TL) approaches, which provide outstanding outcomes in NLP fields in general and abstractive ATS in particular. Transferring the knowledge of large pre-trained models to new datasets is one of the techniques used by TL. These advancements fostered the growth of the field and substantially improved the findings. However, several issues in the field of abstractive ATS remain unresolved, including low levels of novelty and complete coverage of the input article.

2.2 Pretrained Language Models

Large corporations, such as Google, Facebook, and Open AI utilize their resource capabilities to train several Pretrained Language Models (PLMs) on gigantic datasets with specific objectives suitable to diverse Natural Language Processing (NLP) tasks, which are demonstrated in Fig. 1. In general, PLMs are often categorized as encoder-only, decoder-only, and encoder-decoder models. Encoder-only models are most suited for understanding and classification tasks that do not require generating text, such as sentiment analysis and extractive ATS. On the other hand, decoder-only models are optimal for open-ended generation tasks, such as story generation from scratch and text completion tasks. Encoder-decoder models are utilized for sequence-to-sequence tasks in which the output is generated based on the text input, such as machine translation and abstractive ATS. However, warm-started models can be leveraged in sequence-to-sequence tasks by combining encoder-only and decoder-only models to encode the input by the encoder model before being passed to the decoder model to generate the output. Table 1 summarizes PLMs and their practical tasks.

Table 1: PLM Types and Suitable NLP Tasks

PLM Type	**PLM Examples**	**NLP Optimal Tasks**	**NLP Task Examples**
Encoder-Only	Bert [21], Roberta [24]	Understanding Classification	Sentiment Analysis [25]
Decoder-Only	GPT2 [26], GPT3 [27]	Open-ended Generation	Article Generation [28]
Encoder-Decoder	Bart [22], Pegasus [23]	Conditional Generation	Abstractive ATS [29]

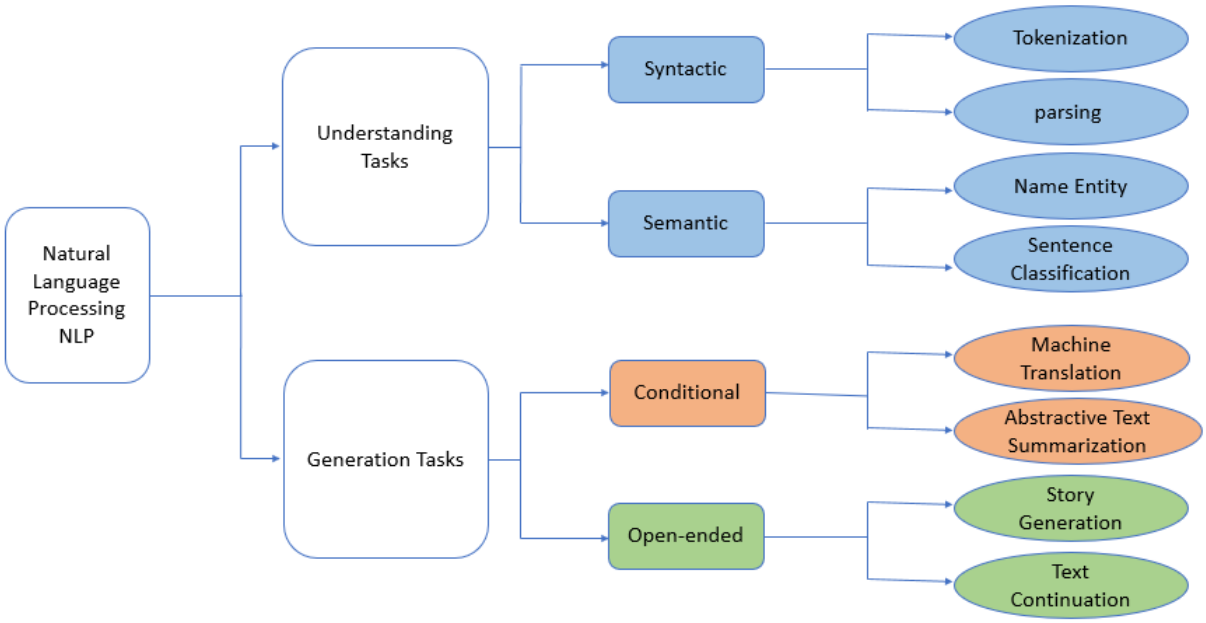

Fig. 1: NLP various Tasks

2.3 Divide and Conquer

Summarizing lengthy materials like scientific papers is one of the most challenging ATS tasks since it requires producing concise yet thorough summaries of the essential components of the large text [30]. The divide and conquer technique approach is utilized in long article text summarization in [31] to cover essential details in long academic articles. The researchers trained their model to divide long documents into chunks, then summarize each section separately before combining them into a comprehensive summary. This model has achieved SotA results in the field of summarizing long documents such as academic articles [10].

2.4 Metrics

2.4.1 Novelty

One of the key reasons for the low-novelty problem is that the training dataset itself contains summary labels with low novelty levels. For example, CNN/DM highlights, i.e., summary labels, tend to be more extractive than abstractive [23]. As a result, models trained on this dataset are obliged to show a preference for extractive over abstractive summaries.

Novelty metrics measure the output summary's non-overlapping words with the input article. Chen and Bansal [32] define novelty metric as demonstrated in Equation 1.

$$M(S,T,n) = \frac{||U(S,n) - U(T,n)||}{||(S,n)||} * 100\% \quad (1)$$

Where M is the novelty metric, U calculates unique words, n is the *n-grams*, S is the resulting summary, T is the input article, and $||X||$ is the number of words in X.

In order to prevent shorter summaries from receiving a higher score, Equation 1 has been normalized by [18] by multiplying it by the length ratio between resulted and reference summaries, as follows:

$$L(S,T,R,n) = \frac{||U(S,n) - U(T,n)||}{||U(S,n)||} * \frac{||S||}{||R||} \quad (2)$$

Where L is the normalized novelty metric, and R is the reference summary. Nonetheless, more extended summaries receive more excellent ratings with this score.

2.4.2 Coverage

One of the most challenging aspects of abstractive ATS is to cover all necessary details throughout the source text. Most abstractive ATS models focus on summarizing specific text parts, especially the beginning while ignoring the remaining details [8]. The training dataset properties, such as CNN/DM, contribute to this bias. Moreover, specific datasets include labeled single-statement summaries for the long source articles, such as Gigaword, XSum, and Reddit_TIFU, resulting in a summary compared to the long input article, which makes it impossible to cover all crucial details in that article.

Coverage is measured by [33] by calculating the fraction of words in the generated summary that are extracted from the source text, as shown in Equation 3.

$$Coverage(A,S) = \frac{1}{|S|} \sum_{f \in F\,(A,S)} |f| \quad (3)$$

where *A* is the input article, *S* is the produced summary, and $f \in F(A, S)$ are all extracted pieces.

However, this measure is inversely proportional to novelty, as it is based on syntactical similarity, i.e., extracted portions.

2.4.3 Rouge

Rouge [34] is the most frequently used metric in ATS research. This metric compares the number of identical words between the generated summary and the highlights. However, this score does not reflect any insight into coverage and novelty. On the contrary, while working on datasets similar to CNN/DM, Rouge scores are proportionally inverse to them. In terms of novelty, the researchers of [23] demonstrate that when the novelty score rises, the ROUGE scores fall. That is because labeled summaries in such datasets tend to be more extractive, indicating low levels of novelty. On the other hand, the researchers of [8] show that CNN/DM's labeled summaries focus on the beginning of the input articles, leaving most of the rest of the article out of consideration; thus, this score delivers lower scores to summaries that cover the entirety of the article.

2.5 Datasets

Training datasets influence the performance of model outputs [35]. For abstractive ATS, numerous datasets with varied criteria are available. CNN/DM contains medium-length news articles along with one- to four-sentence summaries with low levels of novelty. XSum includes medium-length news documents and their corresponding one-sentence summaries with high levels of novelty. Reddit TIFU consists of approximately 123K medium-length online posts accompanied by one short or long summary sentence written with a high degree of novelty [23].

The studies [23] and [10] assessed the datasets based on their extractive coverage and novelty; the results indicate that CNN/DM has a greater degree of extractive coverage but a lower degree of a novelty than XSum and Reddit-TIFU. Intuitively, the extractive coverage level increases as the summary length increases, which explains the dominance of the CNN/DM dataset on XSum and Reddit-TIFU in terms of extractive coverage. Moreover, the novelty levels fall as the number of extracted fragments (i.e., extractive coverage) increases, demonstrating the subservience of CNN/DM in terms of novelty. The specifications of the three datasets are presented in Table 2.

Table 2: Specifications of Datasets used in experiments.

Dataset	Domain	Size	#Words/Input	#Words/Summary
CNN/DM	News	312,085	685.17	51.99
XSum	News	226,711	431.07	23.26
Reddit-TIFU	Social Media	**Long:** 42,984	432.6 (max:500)	23.0 (max: 50)
		Short: 79,949	342.4	9.33 (max: 20)

The researchers of [8] investigated ATS datasets based on the locational bias of crucial information and the learning bias of rigid structural patterns. In contrast to datasets that utilize official news articles, the Reddit-TIFU corpus contains informal postings written by various users along with their summaries. Consequently, models trained on this dataset may not be implicitly biased to learn the tight structural patterns of formal texts in the resulting summaries, thereby avoiding the inclination towards extractive summarization. Consequently, models trained on Reddit-TIFU should generate summaries with higher levels of novelty.

In addition, in formal papers such as news articles, the most important information is located at the beginning of the document. At the same time, the entire material is summarized in the final paragraph. Experiment IV proved this assumption by presenting the ROUGE findings for the predicted summaries of individual article sections, as demonstrated in Section 5.0. In contrast, the key components of informal papers are dispersed throughout. Therefore, in CNN/Daily Mail, the content of the corresponding reference summaries is heavily concentrated on the introduction and conclusion, whereas in Reddit-TIFU, it is dispersed throughout the page [8]. Consequently, models trained on Reddit-TIFU learn how to comprehend the entire text, as opposed to merely locating conclusion sentences.

The researchers of [33] analyzed ATS datasets based on their extractive coverage, density, and compression. In our paper, we define a semantical coverage metric and assess the performance of various models on the CNN/DM dataset in terms of novelty, semantical coverage, and Rouge.

3.0 METHODOLOGY

As discussed in Section 2, existing approaches test their models on the same dataset that they trained their models on. Furthermore, most existing researchers train and test their models on the entire article to generate the final summary. In addition, trained sequence-to-sequence models, such as Pegasus, utilize the same vocabulary and training objective for the encoder and decoder parts. In comparison, our models are evaluated on datasets different than those used in their training. Moreover, the entire article is divided into various parts before being fed into the model to generate distinct summaries for each part, which are then combined to form the final summary in order to enhance coverage. Finally, warm-started models which utilize different encoder and decoder vocabularies and objectives are developed to boost novelty levels.

3.1 Divide and Conquer

This research work aims to increase the degrees of novelty and coverage. Therefore, models trained on datasets with high levels of novelty and generating one sentence per input are utilized. Moreover, to cover all necessary details throughout the entire input article, the divide and conquer approach is used.

In contrast to [31], mentioned in Section 2.3, we utilized the divide and conquer approach only at test time, since at training time we utilized models that had been trained on full articles from other datasets that generate more novelty summaries. Furthermore, the datasets employed in [31] are significantly longer than the datasets utilized in our models. According to our knowledge, this is the first attempt to employ this technique in the medium-length article ATS.

Since CNN/DM labeled summaries are composed initially of bullet points that are concatenated as summaries [6], we can divide the input article into sections, generate a single-sentence summary for each section, and then combine them to form the final summary. Therefore, we trained models on the XSum dataset and utilized models trained on the XSum and Reddit_TIFU datasets that produce a one-sentence summary for each input section. Then we combined these summaries to form the final output.

Mainly first, we divided each input article in the test split of the CNN/DM dataset, which contains 11,490 records, into different numbers of sections in order to conduct different sets of experiments. In particular, the articles are divided into two, three, and four sections. Then, the previously discussed models that are trained on different datasets, i.e., XSum and Reddit-TIFU, are applied to produce the results for each section. Finally, the results of each section are conquered to form the final summary, which is intended to cover all the parts of the article and be written in a more novelty style.

3.2 Learning Models

In this research, we built three warm-started models that leverage Bert, Roberta, and GPT2 and then trained them on XSum and CNN/DM datasets. Table 3 compares the specifications of these PLMs.

Table 3: The specifications of the checkpoints used in our experiments.

Model	Transformer Part	Vocabulary Size	Hidden Size	Hidden Layers	Max Position Embeddings	Filter Size	Parameters
BERT-base	Encoder	30,522	768	12	512	3072	110M
RoBERTa-base	Encoder	50,265	768	12	514	3072	125M
GPT2-base	Decoder	50,257	768	12	1024	-	124M

3.2.1 Models Trained on XSum And Reddit_Tifu

Two warm-started models are trained on the XSum dataset to learn how to generate a sentence that summarizes the entire article using new words. For this purpose, the GPT2 model is leveraged as a decoder for our warm-started models, while Bert and RoBerta are used as encoders, resulting in bert2gpt and roberta2gpt warm-started models. Furthermore, two sequence-to-sequence models finetuned on the XSum and Reddit_TIFU datasets are utilized: Pegasus_XSum and Pegasus_Reddit_TIFU. These models are then leveraged to transfer their knowledge to summarize CNN/DM dataset's articles. The results are compared to those of CNN/DM-trained models in terms of coverage and novelty.

Fig. 2 describes the warm-started models, bert2gpt and roberta2gpt, and explains how they were utilized in the study.

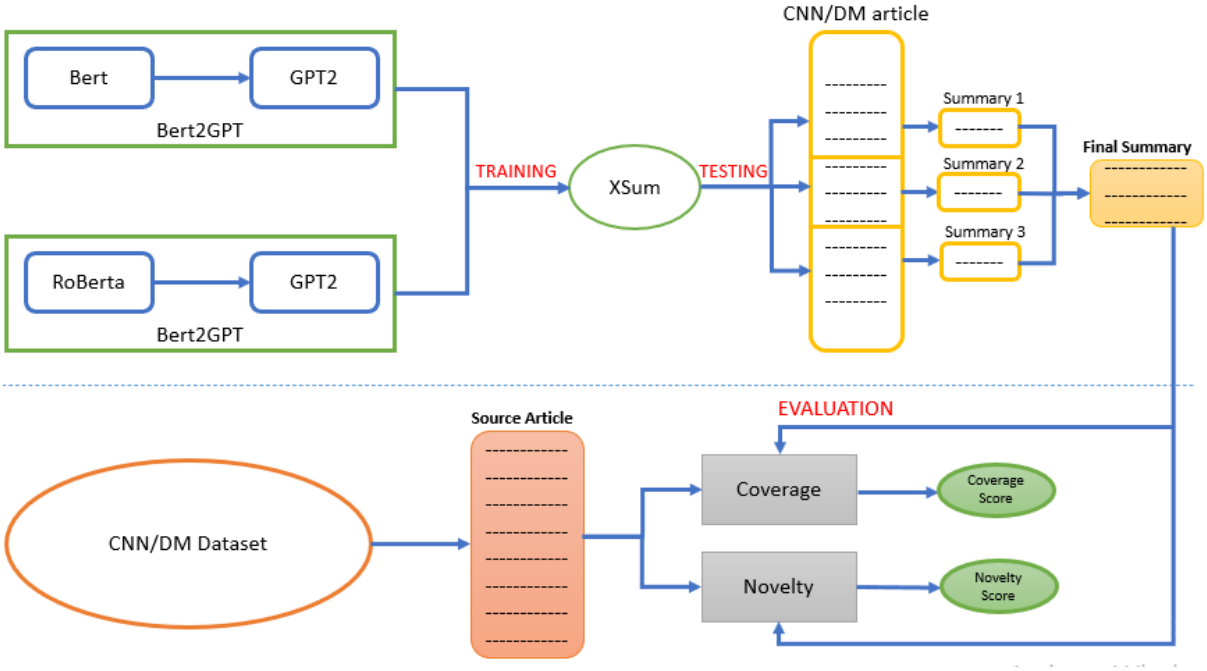

Fig. 2: An explanation of the utilization of warm-started models in this research

3.2.2 Models Trained on CNN/DM

roberta2roberta warm-started model and Pegasus sequence-to-sequence model, which are trained on the CNN/DM dataset, are fed with article sections at test time to compare their performance against models that generate a summary of a complete article input in terms of coverage and Rouge.

Roberta model is utilized as a decoder and encoder in the roberta2roberta model. Roberta has trained to function as an encoder only. However, the researchers of [36] modified encoder-only models to function as decoders as well by applying some adjustments. First, the self-attention layers are adjusted to operate similarly to the decoder in a unidirectional manner. Between the self-attention layer and the two feed-forward layers, a cross-attention layer is introduced. A language model head layer is then built on the decoder's final block. Fig. 3 depicts this model and how it is used in this study.

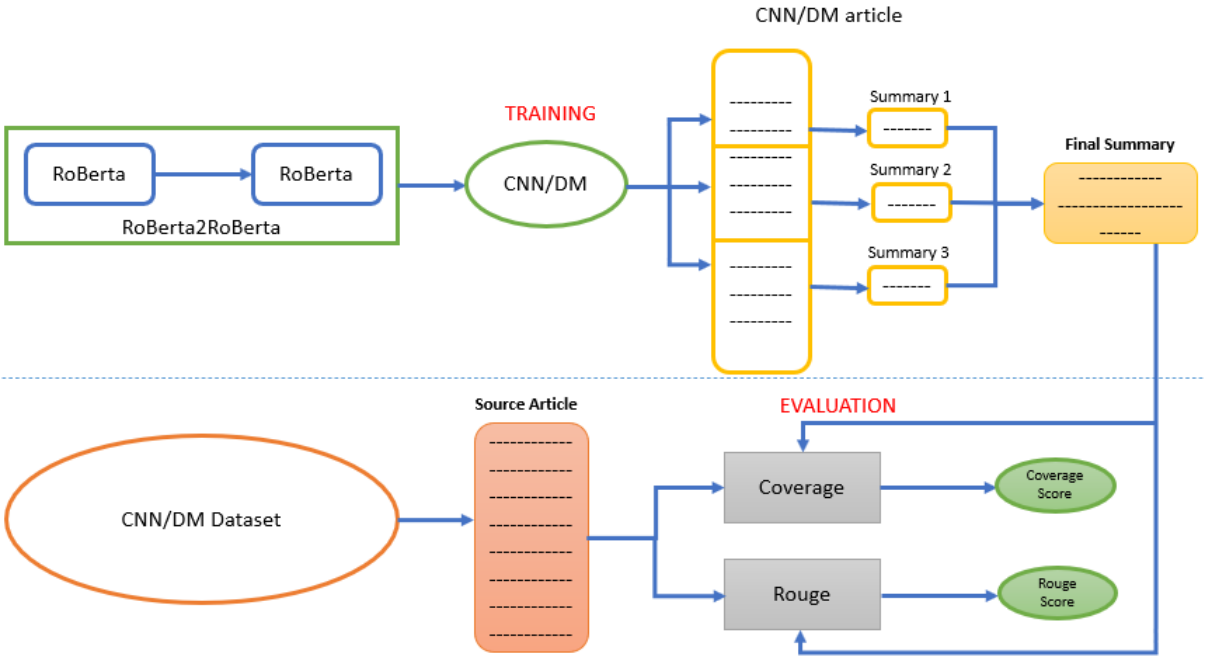

Fig. 3: An explanation of the utilization of roberta2roberta model in this research

4.0 EXPERIMENTAL SETUPS

4.1 Datasets

This study employs three types of abstractive ATS datasets: the non-anonymous version of CNN/DM, XSum, and Reddit-TIFU. As described in Section 2.5, each of these datasets is distinguished by a unique set of characteristics. For instance, CNN/DM is the most popular dataset and is known for its multi-sentence summaries containing 1-4 sentences and approximately 50 words each. XSum and Reddit-TIFU are recognized as having single-sentence summaries written with high levels of novelty [23]. Table 4 analyses the XSUM and CNN/DM datasets in terms of novelty. However, because Reddit-TIFU is written informally by various users, it is recognized to have different writing structures than News article datasets (i.e., CNN/DM and XSum).

Table 4: XSum and CNN/DM n-Gram Novelty

Dataset	1-gram Novelty	2-gram Novelty	3-gram Novelty	4-gram Novelty
XSum	35%	79%	92%	97%
CNN/DM	13%	46%	65%	76%

Based on the characteristics of these datasets, they are utilized in a variety of scenarios and experimental designs in this research.

4.2 Evaluation Metrics

This study employs three kinds of metrics: Novelty, Rouge, and Semantical Coverage.

Rouge is used to estimate the deviation of the proposed model from the Rouge-SotA models. Formally speaking, The ROUGE metric was created to assess the lexical similarity, i.e., n-gram overlapping, between the resulted summary and the reference summary, using three scores, *Recall*, which measures how well the generated summary captures the reference summary, *Precision*, which determines how much of the generated summary is relevant, and *F1*, which computes the harmonic mean of both recall and precision grades. These scores are calculated as follows:

$$Recall = \frac{Num\_overlapping\_words}{Num\_Reference\_summary\_words} \quad (4)$$

$$Precision = \frac{Num\_overlapping\_words}{Num\_resulted\_summary\_words} \quad (5)$$

$$F1 = \frac{2*Recall*Precision}{Recall+Precision} \quad (6)$$

Rouge-F1 is the most employed metric in the majority of abstract ATS studies [10]. This score is used to compute the length of overlapped text between the model-generated summary and the reference summary for uni-grams, bigrams, and the longest sequence (Rouge-1, Rouge-2, and Rouge-LSum, respectively).

Novelty and Semantical coverage are used to measure how human-like is the resulting summary.

To evaluate the novelty levels of each model's summary, we define a new novelty metric to produce 1- to 4-gram novelty scores. As discussed in Section 2.4.1, current novelty equations (Equation 1 and Equation 2) prefer either short or long summaries. To alleviate this behavior, we compute the harmonic mean of these metrics after normalizing them as follows:

$$W(S,T,\mathrm{R},n) = \frac{\left|\left|U(S,n) - U(T,n)\right|\right|}{\left|\left|(S,n)\right|\right|} * \frac{||S||}{||R||} \quad (7)$$

$$K(S,T,R,n) = \frac{\left|\left|U(S,n) - U(T,n)\right|\right|}{\left|\left|(S,n)\right|\right|} * \frac{||R||}{||S||} \quad (8)$$

$$Novelty(S,T,R,n) = \frac{2*W*K}{W+K} \quad (9)$$

Equations 7 and 8 normalize Equation 1 by considering the whole number of tokens in the entire summary, as opposed to merely the number of unique tokens as described in Equation 2. Equation 9 avoids any bias toward either short or long summaries.

In order to compute the semantical coverage levels of models' summaries, similarity measurements between the reference summary and each section are utilized to determine how similar they are. Several similarity measures may be applied for this purpose, including word-based, pairwise, and sentence similarity, which may utilize one of the distances measuring metrics between sentences, such as cosine, Euclidian, and Jaccard [37]. In this work, the *sentence_mpnet_simlarity* model[2], which is based on the MPNet pre-trained model [38], is utilized to determine the semantical similarity between the resulted summary and each section of the input article by computing their sentence embeddings and comparing them using cosine similarity [39].

Specifically, the input article is broken into three sections, and then this metric is utilized to determine the percentage of coverage for each section in the predicted summary, as follows:

$$Semantic\_Coverage(Summary, Section) = sentence\_mpnet\_similarity(Summary, Section) * 100\% \quad (10)$$

4.3 Training Details

We utilized three HuggingFace pre-trained Pegasus checkpoints for the sequence-to-sequence models experiments: Pegasus_CNNDM[3], Pegasus_XSum[4], and Pegasus_Reddit_TIFU[5]. For warm-started models, bert2gpt[6], roberta2gpt[7], and robert2roberta[8], we employed Bert[9] base uncased checkpoint for the encoder part, RoBerta[10] base checkpoint for the encoder and decoder parts, and GPT2[11] base checkpoint is used as a decoder. The utilized models have a hidden size of 768, 12 hidden layers, a 3072 filter size, and 12 attention heads. We utilize an 8-batch train and evaluate the

[2] https://huggingface.co/sentence-transformers/paraphrase-mpnet-base-v2
[3] https://huggingface.co/google/pegasus-cnn_dailymail
[4] https://huggingface.co/google/pegasus-xsum
[5] https://huggingface.co/google/pegasus-reddit_tifu
[6] https://huggingface.co/Ayham/bert_gpt2_summarization_xsum
[7] https://huggingface.co/Ayham/roberta_gpt2_summarization_xsum
[8] https://huggingface.co/Ayham/roberta_roberta_summarization_cnn_dailymail
[9] https://huggingface.co/bert-base-uncased
[10] https://huggingface.co/roberta-base
[11] https://huggingface.co/gpt2

size and the Adam optimizer with betas equal to (0.9,0.999) and epsilon equal to 1e-08. The learning rate is set to 5e-05, warmup steps are set to 2000, and the total finetune epoch is set at 3. For decoding parameters, we employ Beam-Search with a beam size of 4 and remove the duplicated trigrams and 2.0 length penalty during inference. We limit the input articles' length to 512 tokens. For the framework versions, we use Huggingface Transformers 4.12.0.dev0, Pytorch 1.10.0+cu111, Datasets 1.16.3 and 1.18.3, and Tokenizers 0.10.3.

4.4 Baselines

Our models are evaluated in comparison to CNN/DM-trained sequence-to-sequence and warm-started baseline models that have trained on full CNN/DM articles as inputs to produce multi-sentence summaries. These baseline models include Pegasus[12], bert2gpt[13], roberta2rpt[14], and roberta2roberta[15], utilizing different sets of experiments as described in Section 5.0.

5.0 EXPERIMENTAL RESULTS

Various sets of experiments are conducted targeting various objectives. This section discusses the specifics of these experiments and presents their findings, which will be analyzed in depth in the subsequent section.

Experiment I:

In this set of experiments, we compare the performance of warm-started models trained on XSum to models trained on CNN/DM in terms of coverage and novelty. However, based on the discussion in Section 2.4.3, we do not anticipate our proposed models to get high Rouge results. Therefore, we discarded this score for this Experiment set.

In this set of experiments, at test time, the CNN/DM articles are divided into two, three, or four sections before being fed to a model that has been trained to produce a single-sentence summary with high degrees of novelty. As a result, the final summary should consist of two, three, or four sentences that cover all the parts of the input article. Table 5 and Table 6 demonstrate the results.

Table 5: Novelty Results of Various Models Trained on Various Datasets Utilizing Various Divisions

Training Model	Training Dataset	#Sections	Summary Applied on	Model Name	1-g Nov	2-g Nov	3-g Nov	4-g Nov
Divide&Conquer (Our proposed approach)								
Pegasus	Reddit_TIFU	3	Sum(Article/3)	Pegasus_3s_redtifu_cnndm	7%	22%	31%	36%
Pegasus	XSum	2	Sum(Article/2)	pegasus_2s_xsum_cnndm	15%	49%	66%	74%
Pegasus	XSum	3	Sum(Article/3)	pegasus_3s_xsum_cnndm	17%	55%	73%	81%
Pegasus	XSum	4	Sum(Article/4)	pegasus_4s_xsum_cnndm	17%	53%	70%	77%
Bert2gpt	XSum	2	Sum(Article/2)	Bert2gpt_2s_xsum_cnndm	31%	83%	100%	100%
Roberta2gpt	XSum	4	Sum(Article/4)	Roberta2gpt_4s_xsum_cnndm	26%	68%	82%	85%
Baseline Models								
Pegasus	CNN/DM	1	Article	Pegasus_CNNDM	6%	20%	32%	41%
Bert2gpt	CNN/DM	1	Article	Bert2gpt_CNNDM	7%	24%	37%	46%
roberta2gpt	CNN/DM	1	Article	Roberta2gpt_CNNDM	6%	24%	37%	46%
roberta2roberta	CNN/DM	1	Article	Roberta2roberta_CNNDM	2%	20%	36%	48%

Table 6: Levels of Semantical Coverage for Diverse Models Trained on Diverse Datasets Utilizing Various Divisions

[12] https://huggingface.co/google/pegasus-cnn_dailymail
[13] https://huggingface.co/Ayham/bert_gpt2_summarization_cnndm
[14] https://huggingface.co/Ayham/robertagpt2_cnn
[15] https://huggingface.co/Ayham/roberta_roberta_summarization_cnn_dailymail

Training Model	Training Dataset	#Sections	Summary Applied on	Model Name	Sem.Cov Sec1	Sem.Cov Sec2	Sem.Cov Sec3
Divide & Conquer (Our proposed approach)							
Pegasus	Reddit_TIFU	3	Sum(Article/3)	Pegasus_3s_reddit_tifu_cnndm	77.64	69.69	67.32
Pegasus	XSum	2	Sum(Article/2)	pegasus_2s_xsum_cnndm	73.13	63.12	59.98
Pegasus	XSum	3	Sum(Article/3)	pegasus_3s_xsum_cnndm	77.47	67.86	64.19
Pegasus	XSum	4	Sum(Article/4)	pegasus_4s_xsum_cnndm	79.57	69.52	66.36
Bert2gpt	XSum	2	Sum(Article/2)	Bert2gpt_2s_xsum_cnndm	62.70	56.65	54.78
Roberta2gpt	XSum	4	Sum(Article/4)	Roberta2gpt_4s_xsum_cnndm	68.59	62.87	61.46
Baseline Models							
Pegasus	CNN/DM	1	Article	Pegasus_CNNDM	80.83	68.53	64.10
Bert2gpt	CNN/DM	1	Article	Bert2gpt_CNNDM	83.08	72.54	68.53
roberta2gpt	CNN/DM	1	Article	Roberta2gpt_CNNDM	83.69	73.26	69.45
roberta2roberta	CNN/DM	1	Article	Roberta2roberta_CNNDM	80.90	68.98	64.45

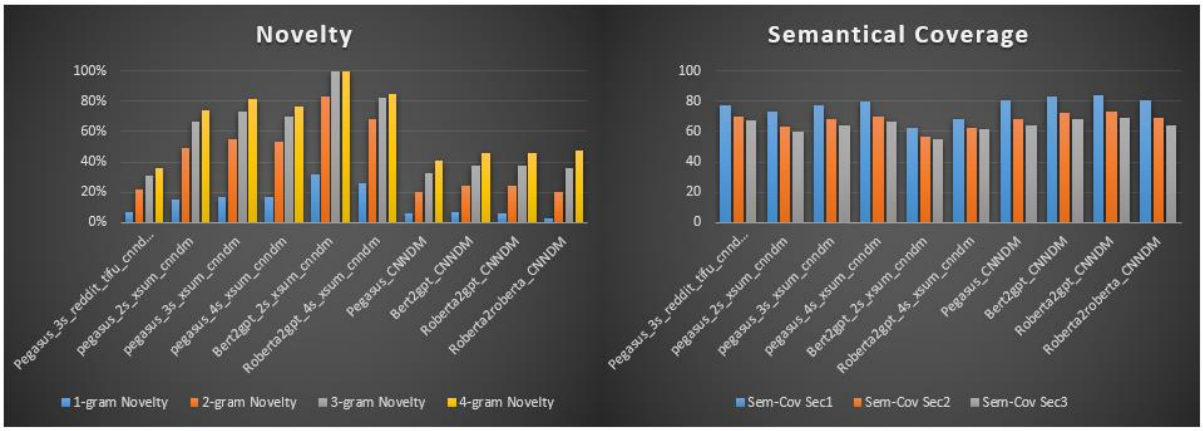

Fig. 4: Levels of Novelty and Semantical Coverage for Diverse Models Trained on Diverse Datasets

Experiment II:

To evaluate the semantical coverage and Rouge scores of CNN/DM-trained models that generate a summary of divided articles as inputs vs. baselines that generate a summary of the complete articles as inputs. However, based on the discussion in Section 2.5, the novelty score is omitted from this set of experiments.

In this set of experiments, the CNN/DM article is broken at the test time into two or three sections and fed to the CNN/DM-trained roberta2roberta model, which provides a one- to four-sentence summary. As a result, the final summary will contain between two and twelve sentences. Hence, for the Experiment of 2-section division, the final summary comprises only the first one/two/three sentences of the first section's summary and the first sentence of the second section's summary. For the Experiment of 3-section division, the final summary combines the summaries generated from each section. Table 7 demonstrates the results.

Table 7: Rouge and Semantical Coverage Results for CNN/DM-Trained Models Utilizing Various Divisions

Training Model	Training Dataset	#Sections	Summary Applied on	Model Name	R-1	R-2	R-L	Sem. Cov Sec1	Sem. Cov Sec2	Sem. Cov Sec3
Our Models										
roberta2roberta	CNN/DM	2	Sum(Article/2)	roberta2roberta_2s_cnndm	41.55	18.97	39.03	81.43	70.73	65.89
roberta2roberta	CNN/DM	3	Sum(Article/3)	roberta2roberta_3s_cnndm	33.86	15.43	32.17	85.01	78.66	77.69

Baseline Models										
Pegasus	CNN/DM	1	Article	Pegasus_CNNDM	44.17	21.47	36.87	80.83	68.53	64.10
Bert2gpt	CNN/DM	1	Article	Bert2gpt_CNNDM	38.09	16.61	35.76	83.08	72.54	68.53
roberta2gpt	CNN/DM	1	Article	Roberta2gpt_CNNDM	39.15	17.68	36.89	83.69	73.26	69.45
roberta2roberta	CNN/DM	1	Article	Roberta2roberta_CNNDM	42.73	19.97	40.09	80.90	68.98	64.45
CNN/DM Dataset					N/A	N/A	N/A	78.21	68.72	65.08

Experiment III:

This set of experiments is merely concerned with measuring novelty degrees based on the newly defined novelty metric, i.e., Equation 9. In this set of experiments, we have not subdivided the input articles into sections to expand semantical coverage. Instead, we feed a model trained on the XSum dataset the predicted summary of a different model that is trained on the CNN/DM dataset, which functions as a paraphrasing process. Moreover, we attempted to generate a CNN/DM article's one-sentence summary that describes the entire input article using a model trained on the XSum dataset. However, as the output of this set of experiments will be a single-sentence summary, we estimate that the semantical coverage and Rouge scores will be quite low; consequently, we omit them.

First, the CNN/DM-trained roberta2roberta model creates a summary, which is then given to the XSum-trained roberta2gpt model to generate a single-sentence summary. Second, the CNN/DM-trained RoBerta2Bert model creates a summary, which is then fed to the XSum-trained bert2gpt model to produce a one-sentence summary. Finally, CNN/DM full articles are summarized in a single sentence by the bert2gpt model, which was trained on XSum. Table 8 shows the results.

Table 8: Novelty degrees for XSum-Trained Models Using CNN/DM predictions and Articles

Training Model	Training Dataset	#Sections	Summary Applied on	Model Name	1-gram Novelty	2-gram Novelty	3-gram Novelty	4-gram Novelty
Our Models								
roberta2roberta_robert a2gpt	XSum	1	Prediction	roberta2roberta_cnndm_roberta2gpt_xsum	32%	86%	100%	100%
roberta2bert_bert2gpt	XSum	1	Prediction	roberta2bert_cnndm_bert2gpt_xsum	33%	79%	93%	95%
Article_Bert2gpt_XSum	XSum	1	Article	CNNDM_Article_Bert2gpt_XSum	33%	88%	100%	100%
Baseline Models								
Pegasus	CNN/DM	1	Article	Pegasus_CNNDM	6%	20%	32%	41%
Bert2gpt	CNN/DM	1	Article	Bert2gpt_CNNDM	7%	24%	37%	46%
roberta2gpt	CNN/DM	1	Article	Roberta2gpt_CNNDM	6%	24%	37%	46%
roberta2roberta	CNN/DM	1	Article	Roberta2roberta_CNNDM	2%	20%	36%	48%

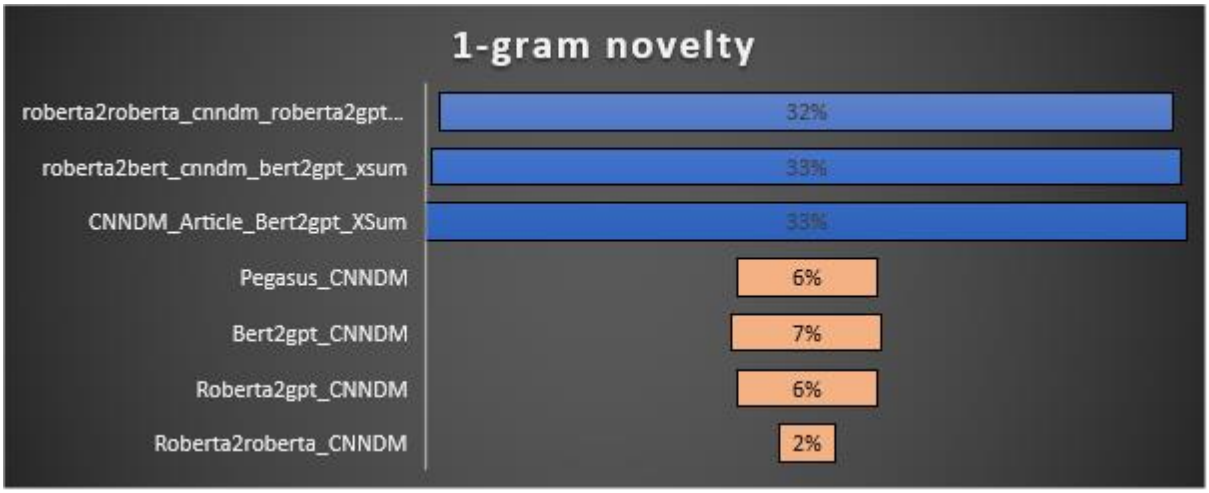

Fig. 5: Results of 1-Gram Novelty metric of Models Trained on XSum Compared to Baselines Trained on CNN/DM

Experiment IV:

This set of experiments is solely concerned with measuring Rouge levels of individual article sections. In this set of experiments, the resulted summary sections are measured in terms of Rouge to determine the extent to which each section influences the ultimate result.

Table 9 presents the outcomes of distinct sections of models applied in Experiment II, whereas Fig. 6 illustrates the disparities between sections' impacts on final output.

Table 9: Rouge Scores for Individual Sections of the CNN/DM-Trained Models' Output

Model Name	Section	Rouge-1	Rouge-2	Rouge-LSum
roberta2roberta_2s_cnndm (Our Model)	Section 1	42.30	19.67	39.68
	Section 2	29.34	8.41	27.03
	Full Summary	41.55	18.97	39.03
roberta2roberta_3s_cnndm (Our Model)	Section 1	41.97	19.46	39.36
	Section 2	30.27	9.25	27.96
	Section 3	27.27	7.07	25.08
	Full Summary	33.86	15.43	32.17

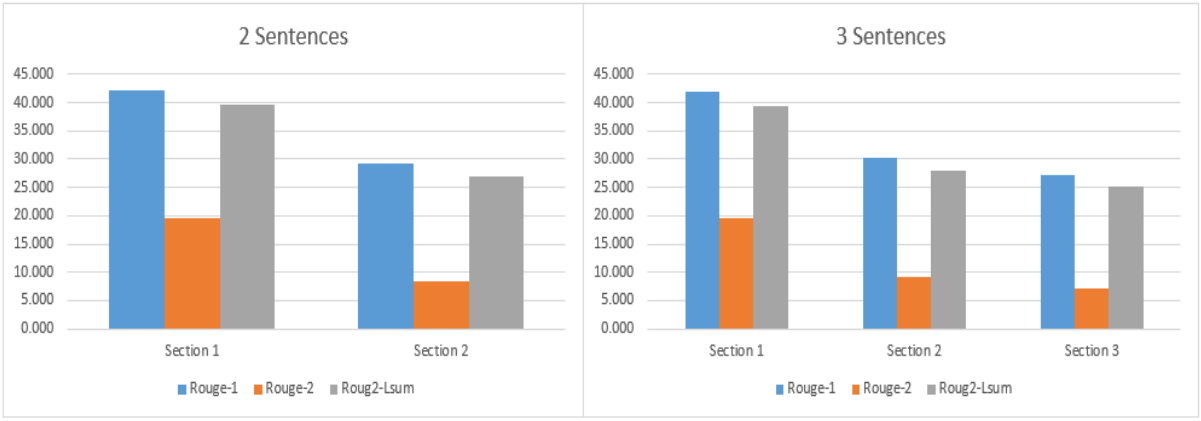

Fig. 6: Sections Effects on CNN/DM-Trained Models' Final Output

6.0 DISCUSSION AND ANALYSIS

For Experiment I, as seen in Fig. 4, generating CNN/DM summaries using models trained on XSum resulted in increased novelty degrees while maintaining equivalent semantic coverage levels in comparison to baseline models.

For Experiment II, as demonstrated in Table 7, baseline models provide summaries that are mainly focus on the article's introduction, which is influenced by the reference summaries of the training dataset, which are given in the last row. This behavior has been alleviated as a result of the divide-and-conquer approach, in which the emphasis has been distributed throughout the three sections of the article.

For Experiment III, as depicted in Fig. 5, the novelty degrees of CNN/DM outcomes are significantly boosted when utilizing models trained on higher novelty degrees, such as XSum.

For Experiment IV, as illustrated in Table 9 and Fig 6, Section 1 received the most attention in the final result summary, followed by the remaining sections in sequence. This behavior is influenced by the structure of the training dataset, which is observed in the final row of Table 7.

Therefore, based on the outcomes of the four sets of studies, we can draw the following conclusions:

Based on Experiment II, the divide-and-conquer approach is capable of improving the semantical coverage of the entire article, rather than focusing on the first section of the article, while generating the summary, as shown in Table 7.

Moreover, according to Experiment III, by utilizing models trained on datasets with higher levels of novelty, the CNN/DM outcomes improved in terms of generating summaries with more novel words that do not appear in the input article.

As a result, based on Experiment I, by employing divide-and-conquer and transferring the learning of models trained on more novel words, the findings show more human-like and concise summaries that cover the majority of the entire input article's key elements with higher novelty levels. However, It is anticipated that models trained on XSum, particularly for the first section, will have less semantical coverage than SotA models trained on full CNN/DM articles, as XSum produces only one sentence compared to three to four sentences that focus on the first section.

In addition, we studied the CNN/DM dataset in terms of semantical coverage and novelty to determine whether these characteristics are influenced more by the dataset itself or the training model itself. First, for the semantical coverage analysis, the reference summaries of the CNN/DM dataset are assessed according to the semantical coverage of each section of their input articles (Table 7). The results indicate that this dataset is biased towards the first article section since it retrieves the most attention when summarizing articles. This behavior is also discussed in [8], as CNN/DM follows the general structure of writing News Articles. As a result, this trait compels models trained on this dataset to produce their results by concentrating on the first section (Experiment IV, Table 9).

Second, for the novelty analysis, Table 4 demonstrates that CNN/DM has lower levels of novelty than XSum. Moreover, Pegasus, one of the SotA models, which is trained on the gap sentence generation objective, generates summaries with high levels of novelty when trained on XSum, but with lower novelty levels when trained on CNN/DM (Table 5). This suggests that the training dataset has a more substantial influence on novelty levels than other factors, including the training objective of the model.

Finally, we evaluated the correlations between the three evaluations used in this work by analyzing the responses of the employed models to them. As depicted in Fig. 7, the Rouge-2 score increases as the semantical coverage increases but declines as the novelty levels increase, whereas the semantical coverage scores decrease as the novelty levels increase. Consequently, the results revealed a tradeoff between Novelty and Rouge and Novelty and Semantical Coverage. Nonetheless, Rouge and Semantic Coverage are proportionally related.

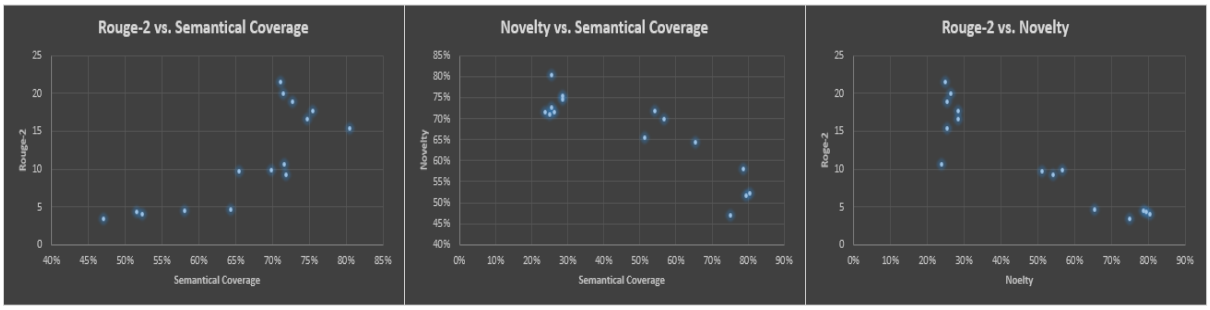

Fig. 7: Relationships between Rouge, Semantical Coverage, and Novelty Metrics

7.0 CONCLUSION

Through the use of divide-and-conquer and transfer learning techniques, the semantical coverage of the entire input and novelty levels are enhanced in this study. Using the divide-and-conquer strategy, for instance, the difference between covering section 1 and the other sections is reduced from 12 and 16 points to 7 and 8 points when comparing the CNN/DM-trained roberta2roberta model. Moreover, the novelty of the summaries is incredibly enhanced by transferring the learning of XSum-trained models to generate CNN/DM summaries. When comparing XSum-trained bert2gpt to CNN/DM-trained bert2gpt, for instance, the 1-gram novelty level increases by 26%. As a result, divide-and-conquer with transferring higher novelty knowledge produced more human-like and concise summaries that covered the bulk of the essential elements of the input article using the model's own words. Moreover, we studied the outcomes to evaluate whether novelty and semantical coverage are impacted more by the dataset itself, as in CNN/DM, or by the training model strategy. The results demonstrate that Pegasus, for example, generates summaries with high levels of novelty when trained on XSum, but with lower levels of novelty when trained on CNN/DM. Furthermore, the results suggest that the reference summaries in this dataset concentrate on the initial section of the article, following the News article writing style of placing the most important details in the first part. Finally, the relationships between metrics are investigated, revealing a tradeoff behavior between novelty and the other two metrics, although Semantical coverage and Rouge have a direct proportion. Therefore, balancing these scores is a research challenge.

ACKNOWLEDGMENT

This work is funded by the Ministry of Higher Education, Malaysia (JPT(BKPI)1000/016/018/25(58)) through Malaysia Big Data Research Excellence Consortium (BiDaREC), via the research grant managed by Universiti Malaya (Grant No.: KKP002-2021).

REFERENCES

[1] N. Hayatin, K. M. Ghufron, and G. W. Wicaksono, "Summarization of COVID-19 news documents deep learning-based using transformer architecture," *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 19, no. 3, pp. 754–761, 2021, doi: 10.12928/TELKOMNIKA.v19i3.18356.

[2] E. Chu and P. J. Liu, "Meansum," *Icml*, vol. 2019-June, pp. 2088–2110, 2019.

[3] D. Graff, J. Kong, K. Chen, and K. Maeda, "English gigaword," *Linguist. Data Consortium, Philadelphia*, vol. 4, no. 1, p.:34, 2003.

[4] C. Napoles, M. Gormley, and B. Van Durme, "Annotated Gigaword," *Proc. Jt. Work. Autom. Knowl. Base Constr. Web-scale Knowl. Extr. (AKBC-WEKEX' 12)*, pp. 95–100, 2012, [Online]. Available: http://dl.acm.org/citation.cfm?id=2391218.

[5] K. M. Hermann *et al.*, "Teaching Machines to Read and Comprehend NIPS 2015," *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, pp. 1693–1701, 2015.

[6] R. Nallapati, B. Zhou, C. dos Santos, Ç. Gulçehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence RNNs and beyond," *CoNLL 2016 - 20th SIGNLL Conf. Comput. Nat. Lang. Learn. Proc.*, pp. 280–290, 2016, doi: 10.18653/v1/k16-1028.

[7] S. Narayan, S. B. Cohen, and M. Lapata, "Do not give me the details, just the summary! Topic-aware convolutional neural networks for extreme summarization," *Proc. 2018 Conf. Empir. Methods Nat. Lang.*

Process. EMNLP 2018, pp. 1797–1807, 2018.

[8] B. Kim, H. Kim, and G. Kim, "Abstractive Summarization of Reddit Posts with Multi-level Memory Networks," *arXiv Prepr. arXiv1811.00783*, 2018.

[9] A. Cohan *et al.*, "A discourse-aware attention model for abstractive summarization of long documents," *NAACL HLT 2018 - 2018 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf.*, vol. 2, pp. 615–621, 2018, doi: 10.18653/v1/n18-2097.

[10] A. Alomari, N. Idris, A. Q. M. Sabri, and I. Alsmadi, "Deep reinforcement and transfer learning for abstractive text summarization: A review," *Comput. Speech Lang.*, vol. 71, no. August 2021, p. 101276, 2022, doi: 10.1016/j.csl.2021.101276.

[11] S. Harabagiu and F. Lacatusu, "Topic themes for multi-document summarization," *SIGIR 2005 - Proc. 28th Annu. Int. ACM SIGIR Conf. Res. Dev. Inf. Retr.*, pp. 202–209, 2005, doi: 10.1145/1076034.1076071.

[12] Y. Ko and J. Seo, "An effective sentence-extraction technique using contextual information and statistical approaches for text summarization," *Pattern Recognit. Lett.*, vol. 29, no. 9, pp. 1366–1371, 2008, doi: 10.1016/j.patrec.2008.02.008.

[13] D. Parveen and M. Strube, "Integrating importance, non-redundancy, and coherence in graph-based extractive summarization," *IJCAI Int. Jt. Conf. Artif. Intell.*, vol. 2015-Janua, pp. 1298–1304, 2015.

[14] Y. Yoshida, J. Suzuki, T. Hirao, and M. Nagata, "Dependency-based discourse parser for single-document summarization," *EMNLP 2014 - 2014 Conf. Empir. Methods Nat. Lang. Process. Proc. Conf.*, pp. 1834–1839, 2014, doi: 10.3115/v1/d14-1196.

[15] P. Ren, F. Wei, Z. Chen, J. Ma, and M. Zhou, "A redundancy-aware sentence regression framework for extractive summarization," *COLING 2016 - 26th Int. Conf. Comput. Linguist. Proc. COLING 2016 Tech. Pap.*, pp. 33–43, 2016.

[16] Y. Zhang, Y. Xia, Y. Liu, and W. Wang, "Clustering sentences with density peaks for multi-document summarization," *NAACL HLT 2015 - 2015 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. Proc. Conf.*, pp. 1262–1267, 2015, doi: 10.3115/v1/n15-1136.

[17] M. Ranzato, S. Chopra, M. Auli, and W. Zaremba, "Sequence level training with recurrent neural networks," *4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc.*, pp. 1–16, 2016.

[18] W. Kryściński, R. Paulus, C. Xiong, and R. Socher, "Improving abstraction in text summarization," *Proc. 2018 Conf. Empir. Methods Nat. Lang. Process. EMNLP 2018*, pp. 1808–1817, 2020, doi: 10.18653/v1/d18-1207.

[19] F. Boutkan, J. Ranzijn, D. Rau, and E. van der Wel, "Point-less: More Abstractive Summarization with Pointer-Generator Networks," *arXiv Prepr. arXiv1905.01975*, 2019, [Online]. Available: http://arxiv.org/abs/1905.01975.

[20] A. Vaswani *et al.*, "Attention is all you need," *Adv. Neural Inf. Process. Syst.*, vol. 2017-Decem, no. Nips, pp. 5999–6009, 2017.

[21] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pretraining of deep bidirectional transformers for language understanding," *NAACL HLT*, 2019.

[22] M. Lewis *et al.*, "BART: Denoising Sequence-to-Sequence Pretraining for Natural Language Generation, Translation, and Comprehension," *arXiv Prepr. arXiv1910.13461.*, 2019, [Online]. Available: http://arxiv.org/abs/1910.13461.

[23] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu, "PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization," *Int. Conf. Mach. Learn.*, pp. 11328–11339, 2020, [Online]. Available: http://arxiv.org/abs/1912.08777.

[24] Y. Liu *et al.*, "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *arXiv Prepr. arXiv1907.11692*, 2019, [Online]. Available: http://arxiv.org/abs/1907.11692.

[25] S. A. Waheeb, N. A. Khan, and X. Shang, “an Efficient Sentiment Analysis Based Deep Learning

Classification Model To Evaluate Treatment Quality," *Malaysian J. Comput. Sci.*, vol. 35, no. 1, pp. 1–20, 2022, doi: 10.22452/mjcs.vol35no1.1.

[26] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language Models are Unsupervised Multitask Learners," *OpenAI Blog*, vol. 1, no. 8, p. 9, 2019.

[27] T. B. Brown *et al.*, "Language Models are Few-Shot Learners," *arXiv Prepr. arXiv2005.14165*, 2020, [Online]. Available: http://arxiv.org/abs/2005.14165.

[28] R. Luo *et al.*, "BioGPT: generative pre-trained transformer for biomedical text generation and mining," *Brief. Bioinform.*, vol. 23, no. 6, pp. 1–12, 2022, doi: 10.1093/bib/bbac409.

[29] M. P. Nguyen and N. T. Tran, "Improving Abstractive Summarization with Segment-Augmented and Position-Awareness," *Procedia CIRP*, vol. 189, pp. 167–174, 2021, doi: 10.1016/j.procs.2021.05.081.

[30] P. C. Yeh, J. Y., Tsai, C. J., Hsu, T. Y., Lin, J. Y., & Cheng, "FEATURE SELECTION AND CLASSIFICATION INTEGRATED METHOD FOR IDENTIFYING CITED TEXT SPANS FOR CITANCES ON IMBALANCED DATA," *Malaysian J. Comput. Sci.*, vol. 34, no. 4, pp. 355–373, 2021, doi: https://doi.org/10.22452/mjcs.vol34no4.3 ABSTRACT.

[31] A. Gidiotis and G. Tsoumakas, "A Divide-and-Conquer Approach to the Summarization of Long Documents," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 28, pp. 3029–3040, 2020, doi: 10.1109/TASLP.2020.3037401.

[32] Y. C. Chen and M. Bansal, "Fast abstractive summarization with reinforce-selected sentence rewriting," *ACL 2018 - 56th Annu. Meet. Assoc. Comput. Linguist. Proc. Conf. (Long Pap.*, vol. 1, no. 2017, pp. 675–686, 2018, doi: 10.18653/v1/p18-1063.

[33] M. Grusky, M. Naaman, and Y. Artzi, "Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies," *NAACL HLT 2018 - 2018 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf.*, vol. 1, pp. 708–719, 2018, doi: 10.18653/v1/n18-1065.

[34] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries.," *Text Summ. Branches out*, pp. 74–81, 2004, doi: 10.1253/jcj.34.1213.

[35] M. J. Yazi, F. S., Vong, W. T., Raman, V., Then, P. H. H., & Lunia, "An Experimental Evaluation of deep neural network model performance for the recognition of contradictory medical research claims using small and medium-sized corpora.," *Malaysian J. Comput. Sci.*, pp. 68–77, 2021.

[36] S. Rothe, S. Narayan, and A. Severyn, "Leveraging pre-trained checkpoints for sequence generation tasks," *Trans. Assoc. Comput. Linguist.*, vol. 8, pp. 264–280, 2020, doi: 10.1162/tacl_a_00313.

[37] Aruna Kumara B and Mallikarjun M Kodabagi, "FEATURE ENGINEERING WITH SENTENCE SIMILARITY USING THE LONGEST COMMON SUBSEQUENCE FOR EMAIL CLASSIFICATION," *Malaysian J. Comput. Sci.*, pp. 65–78, 2022.

[38] K. Song, X. Tan, T. Qin, J. Lu, and T.-Y. Liu, "MPNet: Masked and Permuted Pretraining for Language Understanding," vol. 33, no. NeurIPS, pp. 16857–16867, 2020.

[39] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," *EMNLP-IJCNLP 2019 - 2019 Conf. Empir. Methods Nat. Lang. Process. 9th Int. Jt. Conf. Nat. Lang. Process. Proc. Conf.*, pp. 3982–3992, 2020, doi: 10.18653/v1/d19-1410.

BIOGRAPHY

Ayham Alomari received a master's degree in Computer Science from Yarmouk University, Jordan, in 2010, and he is currently pursuing a Ph.D. degree in Artificial intelligence from the University of Malaya, Malaysia. His research interests are Artificial intelligence, machine learning, natural language processing, text summarization, machine translation, sentiment analysis, deep learning, transfer learning, pre-trained models, and reinforcement learning.

Norisma Idris received a Ph.D. degree in computer science from the University of Malaya in 2011. She joined the Faculty of Computer Science and Information Technology, University of Malaya, in 2001, where she is currently an Associate Professor in the Artificial Intelligence (AI) Department. She is currently working on a few projects, such as Malay Text Normalizer for Sentiment Analysis with industry and Implicit and Explicit Aspect Extraction for Sentiment Analysis under the Research University Grant. For the past five years, she has published more than 15 articles on NLP and AI in various WoS-indexed journals. Her research interest includes natural language processing (NLP), where the main focus is on developing efficient algorithms to process texts and make their information accessible to computer applications, mainly on text normalization and sentiment analysis. She also serves as a reviewer for various journals.

Aznul Qalid Md Sabri received the Erasmus Mundus Master's degree in Vision and Robotics (ViBot) and a joint master's degree from three different universities (University of Burgundy, France; University of Girona, Spain; and Heriot-Watt University, Edinburgh, U.K.), for which he performed in a Research Internship Program at the Commonwealth Scientific Research Organization (CSIRO), Brisbane, Australia, focusing on medical imaging. He is currently a Senior Lecturer with the Department of Artificial Intelligence, Faculty of Computer Science and Information Technology (FCSIT), University of Malaya, Malaysia. The Ph.D. degree (très honorable) on the topic of "Human Action Recognition," under a program jointly offered by a well-known research institution in France, Mines de Douai (a research lab) and the University of Picardie Jules Verne, Amiens, France. He is an active Researcher in the field of Artificial Intelligence, having published in multiple international conferences as well as international journals. His main research interests include the field of computer vision, robotics, and machine learning.

Izzat Alsmadi received master's and Ph.D. degrees in software engineering from North Dakota State University in 2006 and 2008. He is currently an Associate Professor with the Department of Computing and Cyber Security at Texas A&M University, San Antonio. He has more than 100 conferences and journal publications. He is the Lead Author and Editor of several books, including The NICE Cyber Security Framework: Cyber Security Intelligence and Analytics (Springer, 2019), Practical Information Security: A Competency-Based Education Course (Springer, 2018), and Information Fusion for Cyber-Security Analytics—Studies in Computational Intelligence (Springer, 2016). His research interests include cyber intelligence, cyber security, software security, machine learning, natural language processing, software testing, social networks, and software-defined networking.