IJETAE
EXPLORING RESEARCH AND INNOVATIONS

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (E-ISSN 2250-2459, Scopus Indexed, ISO 9001:2008 Certified Journal, Volume 13, Issue 02, February 2023)

Manuscript Received: 02 November 2022, Received in Revised form: 21 January 2023, Accepted: 01 February 2023 *DOI: 10.46338/ijetae0223_07*

OCR-based Hybrid Image Text Summarizer using Luhn Algorithm with FinetuneTransformer Modelsfor Long Document

Van Zachary V. Singco[1], Joel C. Trillo[2], Cristopher C. Abalorio[3], James Cloyd M. Bustillo[4], Junell T. Bojocan[5], Michelle C. Elape[6]

[1,2,3,4,5,6]*Computer Education Department, ACLC College of Butuan, Butuan City, Philippines*
[3,4,6]*Graduate Programs, Technological Institute of the Philippines – Quezon City*

***Abstract*—The accessibility of an enormous number of image text documents on the internet has expanded the opportunities to develop a system for image text recognition with text summarization. Several approaches used in ATS in the literature are based on extractive and abstractive techniques; however, few implementations of the hybrid approach were observed. This paper employed state-of-the-art transformer models with the Luhn algorithm for extracted texts using Tesseract OCR. Nine models were generated and tested using the hybrid text summarization approach. Using ROUGE metrics, we compared the proposed system finetune abstractive models against existing abstractive models that use the same dataset Xsum. As a result, the finetune model got the highest ROUGE score during evaluation; in ROUGE-1 score was 57%, the ROUGE-2 score was 43%, and the ROUGE-L score was 42%. Furthermore, even when better algorithms and models were available for summarization, the Luhn algorithm and T5 finetune model provided significant results.**

***Keywords*—Text Summarization; Extractive; Abstractive; Hybrid; OCR**

I. INTRODUCTION

In recent years, voluminous text data have become accessible on the internet, and the quantity continues to expand dramatically over time, resulting in several issues of information overload. Henceforth, effort and time may save by readers when the focus is on digesting the necessary information and avoiding the unimportant portions upon reading several summarized papers on a particular topic [1][2]. With the advancement of the internet, big data, and relevant technologies, documents are kept in images, resulting in the digitalization of resources in many industries. Recent image processing research has demonstrated the significance of image content retrieval [3][4].

Manual writing, encoding, and summarizing document text images are cumbersome, costly, and time-consuming [5].

The text images provide helpful information, but like frozen text, they inhibit searching and editing a word or sentence, that is why methods for digitizing and summarizing image documents have become sought after by many researchers.

The text summarization techniques which is under Natural Language Processing (NLP) are classified into two: extractive and abstractive methods. The extractive method[6][7][8] concatenates necessary sentences from the original document without modification, while the abstractive process[9][10][11] retains the original document's meaning but creates new phrases. However, extractive encounters issues in the coherence of the sentences that suffer the sense of the summary; meanwhile, abstractive produces human-like output but tend to compromise the semantic information in summary. In literature, the hybrid method combining the two techniques has been proven to provide a good quality text summary performance, especially for different lengths of text documents [12].

In this study, an Optical Character Recognition (OCR) based hybrid image text summarizer usingTesseract with Luhn algorithm and state-of-the-art finetune Transformer models for long documents is implemented. This study is conducted to assist readers in speeding up the process of digitizing and summarizing the image document.

The remaining sectionsin this study are ordered as follows. The literature review presents text summarization, approaches, evaluation metrics, and relevant technology. The framework of the study discusses the steps to implement the text summarizer presented in Section 3. In Section 4, ROUGE metrics results are presented. Finally, in Section 5, we conclude the application as a text summarizer for long documents.

II. Literature Review

A. Text Summarization

Automatic Text Summarization (ATS) system compresses a lengthy text into a more squeezedform by expressing the most essential information in a comprehensible style[13][14]. An ATS presented by Alomari et al. utilized deep neural sequence-to-sequence models and transfer learning (TL), a key topic in Natural Language Processing (NLP)[1]. Researchers are analyzing and developing feasible strategies as ATS applications grow. The methods such as Reinforcement Learning (RL), Deep neural sequence-to-sequence models, and Transfer Learning (TL), including Pre-Trained Language Models, are examples of innovative performance and accuracy in abstractive ATS (PTLMs).As an outcome, generated summaries by PTLMs through universal and deep semantics, as well as word embedding features, show that they can help to increase the quality of the summarized results for the knowledge enhancement/generation guide. Transformers and PTLMs are widely used in numerous NLP tasks, including abstractive ATS, demonstrating their value. Zhang et al. introduced a TextRank[15]method that operates at the corpus level, leveraging resources from available domains based on words and phrases to calculate the word's score in the dataset. They provide a unique strategy for resolving two shortcomings in state-of-the-art in their work. First, there is not onlysingle Automatic Term Extraction (ATE)approach constantly outperforms in all areas. As a result, they are exploring the development of a general process that has the potential to increase the performance of existing (ATE)methods in terms of accuracy. Then, they conclude to typically viable to leverage available lexical resources to assist the ATE or quickly build such resources because unsupervised techniques are the primary approaches common in the ATE. Based on these concepts, they provide AdaText, a general method for updating the TextRank algorithm and applying it to the ATE to improve its accuracy.Jaafar&Bouzoubaa present a new hybrid method for ATS composed of both techniques, extractive and abstractive. They utilized semantic analysis and Conceptual Graphs (CGs) [16]. CGs express the meanings of sentences in a way that humans and computers can both read and understand. Various projects utilized CGs, including intelligence gathering and natural language processing. They begin by obtaining from the input text the salient sentences. With the use of semantic analysis, each sentence generates the corresponding CGs.

These operations will assist in reducing the count of terms and sentences required to create a summary. They are reducing the number of conceptual graphs by working on operations CGs using contractions, joins, and generalization. However, their research result indicates the abstractive method's good performance; they do not dismiss the extractive, which, overall, aids in lessening the complexity by choosing the salient sentences of a text. As in the finishing procedure, new forms of sentences are produced from the CGs. Although they achieve good results with short and simple sentences, it is noteworthy that the combined approach of abstractive and extractive can also be implemented in other languages.

B. Other Approaches in Text Summarization

A Firefly-based text summarizing (FbTS) [17], which uses the firefly algorithm to perform text summarization, was proposed by Tomer& Kumar. According to the researchers, multi-document extraction of automated text summarizing using the ROUGE score improves Firefly algorithm (FA) performance. Moreover, multi-document summarization encounters more issues than summarization in a single document. Their paper presents an FbTSalgorithm method based on Rough Scores, new feature methods, and a fitness function. As a result, to improve performance in multi-document, a novel meta-heuristic-based algorithm for multi-document text summarization approaches is widely used. In multi-document text summarization, there are three factors utilized to generate scores in every phrase: the Topic Relation Factor (TRF), Cohesion (CF), and Readability (RF). This approach is an innovative fitness function according to their suggested technique. The use of TRF, CF, and RF has improved the quality of the extracted summary. Based on their findings, the proposed FbTS algorithm for text summarization with time complexity O (n^2 t) was developed and tested on datasets. They also used a roughness score metric, and the FbTS algorithm assigned higher roughness scores to ROUGE-1 and ROUGE-2.

The Query-focused Sentiment-Oriented Multi-Objective Crowd Search Algorithm [18] (QSO-MOCSA) designed by Sanchez-Gomez et al. solves the issues in extractive multi-document text summarization. This method optimizes objective functions, reduces redundant phrases, and identifies sentiment relevance. The researchers addressed the problem by modifying the metaheuristic population-based crow search algorithm. It calculates ROUGE values and sentiment ratings based on their findings, resulting in statistically evaluated summaries using TAC datasets.

Compared with other approaches, ROUGE scores for ROUGE-1 and ROUGE-2 are 0.4728 and 0.2987, respectively, and the sentiment relevance mean score is 1.9686, which is an outstanding value. ROUGE and sentiment relevance (SR) scores are used to construct summaries and queries of subjects taken in the Opinion Summarization Track of TAC datasets. Furthermore, the results show that QSO-MOCSA outperforms previous approaches, with a 75.5 percent average improvement for ROUGE-1 and 441.3 percent improvement for ROUGE-2.

Tanfouri et al. proposed a genetic method for extractive Arabic text summarization [19]. GA techniques make use of text preprocessing, sentence scoring, and the GA algorithm. Their goal is to select the k most important phrases from D that maximize the fitness function F, where k > n. They put their proposed technique to the test on the EASC Corpus dataset and the Multilingual Corpus, using ROUGE-1 and ROUGE-2 to measure recall, precision, and F-score. According to their findings, the GA approach outperforms previous approaches to Arabic summarizing and aids in the selection of the most.

Du et al. proposed a BioBERTSum[20]model to obtain sentence and token-level contextual representation. A domain-aware bidirectional language model pre-trained on a large-scale biomedical corpus is utilized to employ this model. The pre-trained model is used as an encoder and fine-tuning for the extractive method for single documents. The researchers used a mechanism to identify the position of the information within the sentences. The technique is called sentence position embedding; this is integrated to capture the structural features of the sentence in a document. Based on their results, the BioBERTSum model outperforms the most recent SOTA (state-of-the-art) model to the ROUGE-1, ROUGE-2, and ROUGE-L on the PubMed dataset. They aim to select the essential pre-trained language model, the “SOTA” (state-of-the-art) and fine-tune it for extractive summarization in the biomedical domain. Meaningful sentences while avoiding summary redundancy. They also used the GA method to obtain salient sentences in their final summary by maximizing a specific fitness function and arriving at the best answer.

Tawmo et al. proposed the Text-to-Text Transformer (T5) Transformer model [21] for the Abstractive text summarization method, to analyze the performance datasets on the CNNDM, MSMO, and XSUM; and, to compare the output test results on the datasets to determine the model's and dataset’s ability in terms of ROUGE and BLEU scores. They calculated and analyzed the use of three distinct datasets to test, train, and validate the CNNDM, MSMO, and XSUM datasets.

The CNNDM dataset has 2,868,171 documents on the train model result, 13,368 documents in the test result, and 11,490 documents in the validation test result. The MSMO dataset then contains 2,936,625 article sources and data on the train model result; their test result includes 10,295 articles as well as the validation of 10,339 articles. They also tested the Extreme Summarizing (XSUM) dataset, which contains 204,045 train model results and a test result of 11,334 with a validation test result of 11,332. Calculate the overall dataset using the three (3) proposed models: CNNDM, MSMO, and XSUM. When compared to the two datasets of CNNDM and MSMO, which have higher dataset results, the XSUM datasets with a minimum of 90% of train data and 5% of data validation and 5% of test data have the excellent achieved performance. As a result, the sentence-level BLEU score on the three datasets shows a majority similarity of 25 to 50 points. In addition, the ROUGE score ranges from 25 to 50 for sentence-level similarity. Therefore, their evaluation results demonstrate that the MSMO dataset of ROUGE-1 score has a higher 42.287 compared to each data of ROUGE-2 and ROUGE-L on the CNNDM and XSUM; the ROUGE score and also the BLEU score have a higher of 43.9 compared to each data of BLEU-2, BLEU-3, BLEU-4, and BLEU data on CNNDM and XSUM. As a result, it was found by ROUGE score and BLEU score predictions that the T5 model produces the best results with the MSMO dataset. Their findings revealed that the pre-trained T5 transformer model has a short abstract result summary that is excellent, concise, fluent, and coherent.

Suleiman &Awajan proposed using deep learning model techniques [22] in addition to the datasets they already had. Based on the results, the ROUGE 1, ROUGE 2, and ROUGE L were split into clusters. The first cluster deliberated using Single-Sentence Summary techniques, while the second cluster used Multisentence Summary techniques. They are measures to use and evaluate the quality of summarization, and the ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metrics of ROUGE 1, ROUGE 2, and ROUGE L have been identified to be frequently used. The issues encountered during the summarization procedure, as well as the proposed solutions for each approach, are examined. As a result, they examined the most recent techniques, datasets, evaluation metrics, and challenges for these applications using deep learning techniques in abstractive text summarization and produced excellent results. During their research, they tested every possible combination in order to present the most widely used technique for abstractive text summarization, which is (LSTM).

They use deep learning models to assess the evaluation results on documents from the CNN/Daily Mail datasets. Based on their evaluation metric results, it shows that some approaches with pretrained encoder models over the shows the most dominantmethods for abstractive text summarization and have the highest achieved value of 43.85 in ROUGE 1, a value of 20.34 in ROUGE 2, and a value of 39.9 in ROUGE L when it comes to generating summary; and also the most promising feature of all, with the use of BERT word embedding, which is based on the Transformer.

Furthermore, they used abstractive summarization to recognize the significance of the text, to produce a shorter forms that best expresses the meaning, and to determine that the most abstractive text summarization models encounteredissues.

C. Text Summarization Evaluation Metrics

ROUGE-AR, a standard ROUGE metric proposed by Maples [23] extends ROUGE to include both AR and a measure of summary readability. "AR" refers to Anaphora Resolution is a type of pronoun resolution; in their model, they implement the RNN encoder-decoder provided in. To extract data and preprocess the words, they used an English Gigaword Fifth Edition corpus. They used the DUC summaries (2002) to evaluate the ROUGE variant. Among the three (3) ROUGE 1, ROUGE 2, ROUGE 3, and ROUGE L metrics for evaluation in the generated summary, the ROUGE-AR has the highest achieved performance of 0.108 to ROUGE-1, 0.106 to ROUGE-2, 0.1 to ROUGE-3, and 0.07 to ROUGE-L. However, during the evaluation metric's results testing, they discovered that the metric's parameters could have been fine-tuned more effectively in terms of summary and text quality performance.

D. Optical Character Recognition

In 1920 the OCR device was initiated, and it was patented in Germany in 1929 as a “Reading Machine” by an Austrian engineer Gustav Taucheck[24]. It was seconded in 1933 as “Statistical Machine,” an OCR device patented by Paul Haden of the USA. OCR nowadays efficiently extracts text images. However, its early versions limited obtained texts [25]. The “omni-front” OCR device can recognize any font by Ray Kurzweil [26]. Nowadays, most fonts are recognizable with a high degree of accuracy being offered by OCR systems [27]. OCR uses preprocessing techniques for efficient character recognition, such as deskewing, binarization[28], line removal [29], zoning, and line and word detection.

The OCR technology has been very useful in many fields such as spelling corrections [30], generating course advisory in College [31], and AI for Healthcare[32] as it digitized text images needed for researches. A study by Hubert et al. created a technique for detecting whether or not a picture includes adverts they focused their analysis on detecting promotional offers made available by firms on social media[33]. Optical Character Recognition (OCR) and the Nave Bayes Algorithm will almost surely uncover text in such photos. In this work, they employed datasets in the form of pictures. They must first preprocess the text taken from the photographs before they can utilize it. They then employ Nave Bayes, K-Nearest Neighbors (KNN), and Random Forest to identify whether or not the data includes promotion.

To increase the performance of the Nave Bayes classifier, the data must first be adequately prepared. It has the most influence of all of them. This was proved in their investigation, showing the algorithm's reliability. They employed the validation strategy to divide each picture into five groups in order to train and test their model. The Nave Bayes model exhibited the greatest accuracy of the three algorithms, with 94.31 percent accuracy, 94.33 percent recall, 94.110 percent precision, and a 0.93 FI score. Optical Character Recognition (OCR) and the Nave Bayes Algorithm yield the highest results and deliver the biggest accuracy increase, improving accuracy from 75% to 94.31 percent, according to their findings.

Adjetey&Sarpong proposed a novel algorithm for recognizing text [34], making documents editable and searchable in images, extracting text from images using Levenshtein Algorithm and Tesseract OCR to searchtext from images to find in the document. Begin by locating and comparing the texts extracted from the images using the Levenshtein text-matching algorithm. They also use two main techniques, which are keyword searches and image or picture searches to reverse image search engines. Therefore, based on their analysis, Tesseract OCR indicates that the accuracy level of the 609 words that were uploaded for the test were recognized in the 2167 x 3064 pixels. This is so that some punctuation words that OCR knows could be recognized with less words. This line lines up with the work that was done. With the use of the OCR data, it can be seen that meaningful words can also be produced at high resolution. Additionally, they employed the Levenshtein Algorithm to collect the query image with the highest comparison ratio of 1. It demonstrates that the greatest ratio of 0.88 and 0.22 test results are a perfect fit. With these outcomes, it is evident that the comparison ratio rises as the number of words in the query image increases.

III. METHODOLOGY

This section discussed on how the method is implemented as shown in Fig. 1.

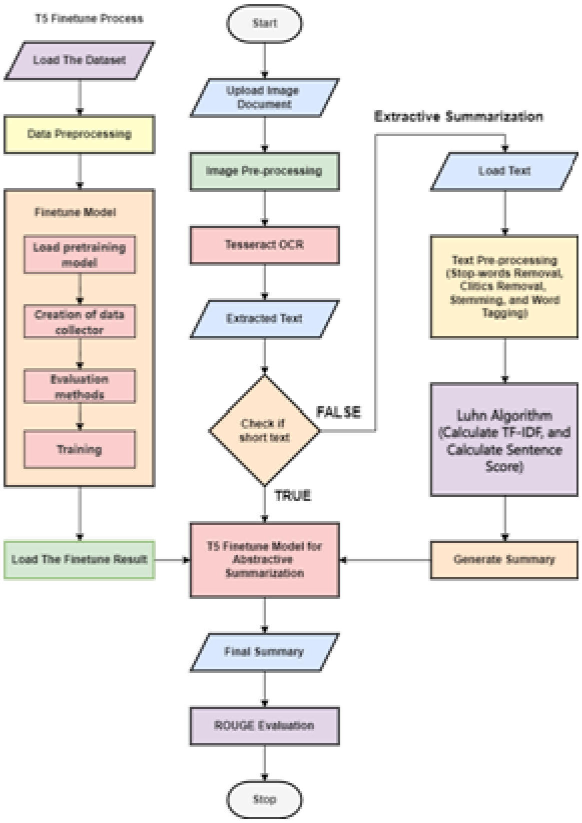


Fig 1.Framework of the study.

The steps in the framework of the study are as follows:

A. Upload Image Documents

The text summarizer application uploads the text image document to the system containing the text to be summarized.

B. Image Pre-processing

Some uploaded image documents have a lot of noise. As a result, characters are difficult to recognize. The noise can be decreased by using preprocessing techniques. Smoothing and normalizing are applied to the image. Filling and thinning methods are used to create an image.

C. Tesseract OCR

The OCR system uses the Tesseract algorithm to recognize characters from image foreground pixels, often known as blobs, and to recognize lines. Following that, these lines are identified as words or characters. In this phase, the image is turned into a character stream, which represents letters.

D. Extracted Text

The output text of the Tesseract OCR are digitized text extracted from the text image.

E. Check If Short Text

This will determine whether the text extracted from the Tesseract OCR is short or long documents. In this hybrid approach, the application executes the extractive technique first then the result will run thru the abstractive process.

F. Load Text

This will load the text from the extracted text.

G. Text Pre-processing

To handle the document, the preprocessing function uses NLTK methods like tokenization, stemming, POS tagger, and stop-words. The preprocessing part divides the text into a list of terms using tokenization functions once the document is input into the program. There are two types of tokenization functions: sentence tokenization and word tokenization. The purpose of sentence tokenization is to divide a paragraph into sentences. Word tokenization, on the other hand, is a function that splits a string of written language into words and punctuation.

H. Calculate TF-IDF with Luhn Algorithm

The Luhn algorithm[35] is a method applied in text summarization. It comprised feature vectorizing techniques such as TF-IDF weighting schemes commonly used in text classification task like sentiment analysis[36] and document classification[37][38][39]. The TF-IDF value of each noun and verb may then be determined using the preprocessed list of words. The TF-IDF equation is shown below. The TF-IDF value spans from 0 to 1 with a ten-digit precision. These words are arranged in decreasing order by their value once computed. The information is then collected into a new dictionary of words and their meanings. This is necessary for analyzing the rank of the TF-IDF value from all of the terms.

$$TF = \frac{Total\ appearance\ of\ word\ in\ documents}{Total\ words\ in\ a\ document} \tag{1}$$

$$IDF = log\,\frac{All\ Document\ Number}{Document\ Frequency} \tag{2}$$

$$TF - IDF = TF\ x\ IDF \tag{3}$$

I. Sentence Score

This will calculate the importance value of a sentence. The total of the significance values of all nouns and verbs in a sentence is the word's importance value. We use the scoring method as illustrated below.

$$Score = \frac{Total\ Score\ per\ Sentence}{Count\ Words\ \ in\ Sentence} \tag{4}$$

J. Generate Summary

This will generate the summary of the original text document. These final sentences have been arranged in the order in which they appeared in the original document.

K. T5 Transformer Model

The generated summary from the extractive summarization will be fed to the T5 transformer model, which is an abstractive summarization algorithm. Rather than collecting sentences directly from the original text, it will rewrite them as necessary.

L. Final Summary

This will be the final output of the extracted text from the image document, which is shorter and possibly includes new phrases and sentences not found in the original text.

M. Finetune Process

First, the xsum dataset is loaded to finetune the T5 transformer model. Then, data will be transformed into tokens using the tokenizer. The tokenizer turns text into tokens such as words or parts of words, punctuation marks, etc. Next step, the T5 pre-trained model is loaded into the process. Then, the data collector is created, which creates a batch by taking a list of dataset components as input. These items are the same as train dataset or evaluation dataset elements. The evaluation method will check the model's performance on a given dataset, typically by comparing the model's predictions to specific ground truth labels. The T5 Transformer model will be trained and optimized. Finally, the finetune model result will be loaded to the T5 transformer.

N. Evaluation Method

The ROUGEevaluation metrics is the most used methodto measure the summary.

The primary goal of ROUGE is to computeoverlapping units between the reference summaries and the system's generated summary. We evaluated three types of ROUGEs to examine the system-generated summary. The three rogues are as follows:

1. ROUGE-1

Precision and Recall measure the similarity of uni-grams in reference and candidate summaries. By uni-grams, we basically mean that each token of comparison is a single word.

2. ROUGE-2

Precision and Recall measure the similarity of bi-grams in reference and candidate summaries. By bi-grams, we mean that each token of comparison is made up of two continuous words from the reference and candidate summaries.

3. ROUGE-L

Precision and Recall calculates the longest common subsequence (LCS) words between reference and candidate summaries. Word tokens that are in sequence but not necessarily consecutive are referred to as LCS.

IV. RESULTS AND DISCUSSION

There are two parts of the evaluation performed in this study. In part 1, to enhance the model's summarization, we trained it on the xsum dataset and compared it to the standard T5 transformer dataset. The assessment measure uses the ROUGE scoring system, with ROUGE-1, ROUGE-2, and ROUGE-L scores considered.

Based on the ROUGE score in Table 1, the Model no. 2 has the highest ROUGE score. It can be determined that the ROUGE-1 score of the Model no. 2 is 23%, the ROUGE-2 score is 26%, and the ROUGE-L score is 18% higher than the Model no. 1. At the same time, the result of Model no. 4 in the ROUGE-1 score is 25%, the ROUGE-2 score is 23%, and the ROUGE-l score is 28 percent higher than the Model no. 3. Therefore, it can be estimated that the ROUGE score of the Model no. 4 orT5 Small + Finetune with xsummodel has been satisfactory.

TABLE 1.
MODEL SCORING RESULTS PART 1

MODEL NO.	MODEL NAME	ROUGE-1	ROUGE-2	ROUGE-L
1	T5 Base	0.3199	0.0338	0.3199
2	T5 Base + Finetune with xsum	0.5499	0.2926	0.4999
3	T5 Small Model	0.2592	0.0344	0.1851
4	T5 Small + Finetune with xsum	0.5128	0.2631	0.4615

The second part of the simulation was performed, and the results are displayed in Table 2. T5 Finetune model is compared with the various entry variants.

Google/pegasus-xsum, Facebook/bart+large+xsum, sshleifer/distilbart+xsum-12-6, and pki/t5+small+finetune with xsum are the models.

TABLE 2.
MODEL SCORING RESULTS PART 2

MODEL NO.	MODEL NAME	ROUGE-1	ROUGE-2	ROUGE-L
5	google/Pegasus + xsum	0.0689	0	0.0689
6	facebook/bart + large + xsum	0.0571	0	0.0571
7	sshleifer/distilbart + xsum-12-6	0.0606	0	0.0606
8	pki/t5 + small + Finetune with xsum	0.2068	0.0689	0.2068
9	Luhn Algorithm + T5 Finetune with xsum	0.5789	0.4324	0.4210

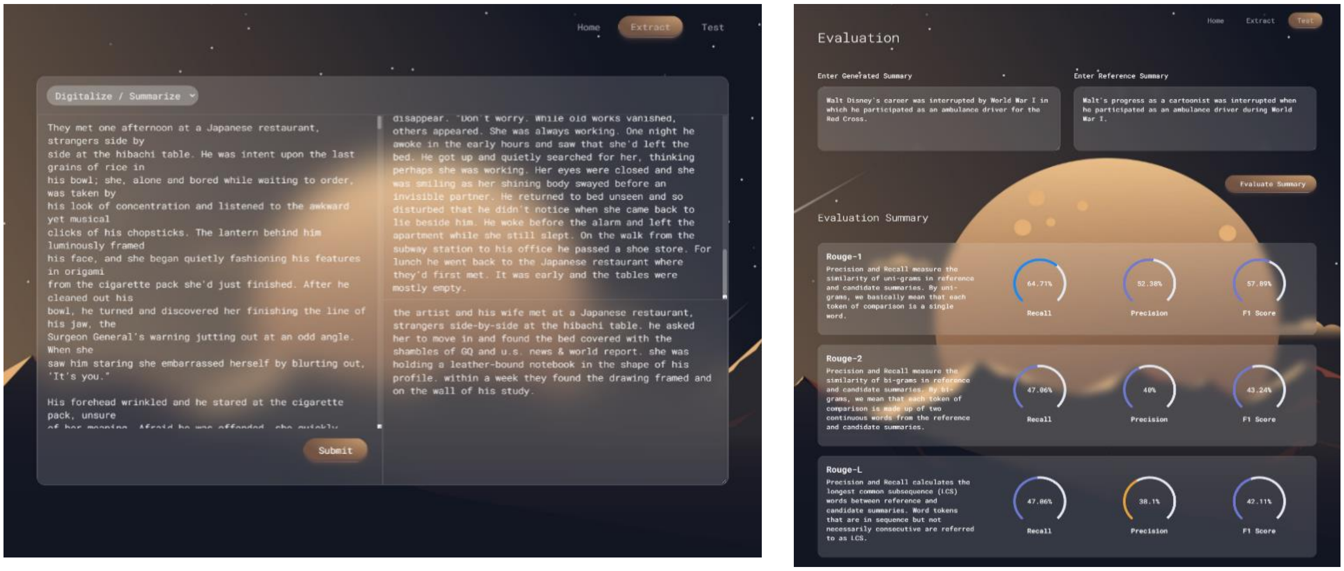
According to Table 2, the approach achieves the best performance for most evaluation scores. Even for the ROUGE-1, ROUGE-2, and ROUGE-L scores, the T5 finetune model has a remarkable result of outperforming the compared models because the model was trained with 1000 steps

Based on Table 2the researcher compared the T5 Finetune model with another model Finetune with the same dataset xsum. Using the sample data in the xsum dataset. The researcher uses the T5 Finetune model to summarize the original text, along with Google/Pegasus + xsum, Facebook/bart+large+xsum, Shleifer/distilbart+xsum-12-6, and Pki/t5+small+Finetunexsum. The generated summaries from the models are then compared with the human summaries using rouge, which calculates the overlap rate of the summary produced by the generated and human summaries.

As a result, the Model no. 9 or theLuhn Algorithm + T5 Finetune + xsum model gets the best results for most evaluation scores. The ROUGE-1 score at 37%, the ROUGE-2 score at 36%, and the ROUGE-L score at 21% are higher than compared Model no. 8.Following the investigation of the results, it is clear that this reason depends on the qualities of the generated summary and scoring techniques in ROUGE. Therefore, the Luhn Algorithm + T5 Finetune + xsum model outperforms the compared models with impressive results.

The original text, the human summary, and the results generated summary evaluated are obtained from theXsum dataset by Google/Pegasus+xsum, facebook/bart+large+xsum, sshleifer/distilbart+xsum-12-6, pki/t5+small+finetune with xsum, and T5 Finetune model.

In addition, we provide a list of reference summaries in Table 3. The names and positions of the people who summarized the original text are listed.



They met one afternoon at a Japanese restaurant,
strangers side by
side at the hibachi table. He was intent upon the last
grains of rice in
his bowl; she, alone and bored while waiting to order,
was taken by
his look of concentration and listened to the awkward
yet musical
clicks of his chopsticks. The lantern behind him
luminously framed
his face, and she began quietly fashioning his features
in origami
from the cigarette pack she'd just finished. After he
cleaned out his
bowl, he turned and discovered her finishing the line of
his jaw, the
Surgeon General's warning jutting out at an odd angle.
When she
saw him staring she embarrassed herself by blurting out,
'It's you."

His forehead wrinkled and he stared at the cigarette
pack, unsure
of her meaning. Afraid he was offended, she quickly

disappear. 'Don't worry. While old works vanished,
others appeared. She was always working. One night he
awoke in the early hours and saw that she'd left the
bed. He got up and quietly searched for her, thinking
perhaps she was working. Her eyes were closed and she
was smiling as her shining body swayed before an
invisible partner. He returned to bed unseen and so
disturbed that he didn't notice when she came back to
lie beside him. He woke before the alarm and left the
apartment while she still slept. On the walk from the
subway station to his office he passed a shoe store. For
lunch he went back to the Japanese restaurant where
they'd first met. It was early and the tables were
mostly empty.

the artist and his wife met at a Japanese restaurant,
strangers side-by-side at the hibachi table. he asked
her to move in and found the bed covered with the
shambles of GQ and u.s. news & world report. she was
holding a leather-bound notebook in the shape of his
profile. within a week they found the drawing framed and
on the wall of his study.

Evaluation

Enter Generated Summary

Walt Disney's career was interrupted by World War I in which he participated as an ambulance driver for the Red Cross.

Enter Reference Summary

Walt's progress as a cartoonist was interrupted when he participated as an ambulance driver during World War I.

Evaluation Summary

Rouge-1

Precision and Recall measure the similarity of uni-grams in reference and candidate summaries. By uni-grams, we basically mean that each token of comparison is a single word.

64.71% Recall
52.38% Precision
57.89% F1 Score

Rouge-2

Precision and Recall measure the similarity of bi-grams in reference and candidate summaries. By bi-grams, we mean that each token of comparison is made up of two continuous words from the reference and candidate summaries.

47.06% Recall
40% Precision
43.24% F1 Score

Rouge-L

Precision and Recall calculates the longest common subsequence (LCS) words between reference and candidate summaries. Word tokens that are in sequence but not necessarily consecutive are referred to as LCS.

47.06% Recall
38.1% Precision
42.11% F1 Score

Fig. 2.Application's summarization and evaluation (left-to-right image)

V. Conclusion

In this paper, the researchers developed the OCR-based hybrid image text summarizer application using the Luhn algorithm with finetuning transformer models, as shown in Fig. 2. To achieve the average ROUGE score of 40% in summarization, researchers need to finetune the T5 transformer model and adjust some hyperparameters during training; as a result, the finetune model got the highest ROUGE score during evaluation, in ROUGE-1 score is 57%, ROUGE-2 score 43% and ROUGE-L score 42%. And also, to address the abstractive summarization of long-term dependencies while dealing with huge texts, researchers used a hybrid approach that combined the extractive and abstractive techniques. To generate a summary, the extractive method extracts the significant sentences from the original text, and the abstractive rewrite the summary phrases based on numerous extracted sentences. It overcomes the shortcomings of prior abstractive approaches when working with a lengthy text document.

Furthermore, even when better algorithms and models were available for summarization, the Luhn algorithm and T5 finetune model provided significant results. Additionally, researchers believe the study can be improved with future development and a thorough understanding of the overall context.

Finally, the researchers consider that the application implemented with the study is helpful to students and readers who want to speed up the digitalizing and summarizing of image documents.

REFERENCES

[1] 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.

[2] K. Hazra et al., "Sustainable text summarization over mobile devices: An energy-aware approach," Sustain. Comput. Informatics Syst., vol. 32, no. August, p. 100607, 2021, doi: 10.1016/j.suscom.2021.100607.

[3] C. Kaundilya, D. Chawla, and Y. Chopra, "Automated text extraction from images using OCR system," in Proceedings of the 2019 6th International Conference on Computing for Sustainable Global Development, INDIACom 2019, 2019, pp. 145–150.

[4] J. Mei, A. Islam, A. Moh'd, Y. Wu, and E. Milios, "Statistical learning for OCR error correction," Inf. Process. Manag., vol. 54, no. 6, pp. 874–887, 2018, doi: 10.1016/j.ipm.2018.06.001.

[5] G. C. V. Vilca and M. A. S. Cabezudo, "A study of abstractive summarization using semantic representations and discourse level information," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2017, vol. 10415 LNAI. doi: 10.1007/978-3-319-64206-2_54.

[6] B. Mutlu, E. A. Sezer, and M. A. Akcayol, "Candidate sentence selection for extractive text summarization," Inf. Process. Manag., vol. 57, no. 6, p. 102359, 2020, doi: 10.1016/j.ipm.2020.102359.

[7] M. Mojrian and S. A. Mirroshandel, “A novel extractive multi-document text summarization system using quantum-inspired genetic algorithm: MTSQIGA,” Expert Syst. Appl., vol. 171, no. November 2019, p. 114555, 2021, doi: 10.1016/j.eswa.2020.114555.

[8] A. Joshi, E. Fidalgo, E. Alegre, and L. Fernández-Robles, “SummCoder: An unsupervised framework for extractive text summarization based on deep auto-encoders,” Expert Syst. Appl., vol. 129, pp. 200–215, 2019, doi: 10.1016/j.eswa.2019.03.045.

[9] Y. K. Atri, S. Pramanick, V. Goyal, and T. Chakraborty, “See, hear, read: Leveraging multimodality with guided attention for abstractive text summarization,” Knowledge-Based Syst., vol. 227, p. 107152, 2021, doi: 10.1016/j.knosys.2021.107152.

[10] Y. Guan, S. Guo, R. Li, X. Li, and H. Zhang, “Frame Semantics guided network for Abstractive Sentence Summarization[Formula presented],” Knowledge-Based Syst., vol. 221, 2021, doi: 10.1016/j.knosys.2021.106973.

[11] R. Bhargava, Y. Sharma, and G. Sharma, “ATSSI: Abstractive Text Summarization Using Sentiment Infusion,” Procedia Comput. Sci., vol. 89, pp. 404–411, 2016, doi: 10.1016/j.procs.2016.06.088.

[12] R. Bhargava and Y. Sharma, “Deep Extractive Text Summarization,” in Procedia Computer Science, 2020, vol. 167. doi: 10.1016/j.procs.2020.03.191.

[13] W. S. El-Kassas, C. R. Salama, A. A. Rafea, and H. K. Mohamed, “Automatic text summarization: A comprehensive survey,” Expert Syst. Appl., vol. 165, p. 113679, 2021, doi: 10.1016/j.eswa.2020.113679.

[14] A. P. Widyassari et al., “Review of automatic text summarization techniques & methods,” Journal of King Saud University - Computer and Information Sciences. King Saud bin Abdulaziz University, 2020. doi: 10.1016/j.jksuci.2020.05.006.

[15] Z. Zhang, J. Petrak, and D. Maynard, “Adapted textrank for term extraction: a generic method of improving automatic term extraction algorithms,” in Procedia Computer Science, 2018, vol. 137, pp. 102–108. doi: 10.1016/j.procs.2018.09.010.

[16] Y. Jaafar and K. Bouzoubaa, “Towards a New Hybrid Approach for Abstractive Summarization,” Procedia Comput. Sci., vol. 142, pp. 286–293, 2018, doi: 10.1016/j.procs.2018.10.496.

[17] M. Tomer and M. Kumar, “Multi-document extractive text summarization based on firefly algorithm,” J. King Saud Univ. - Comput. Inf. Sci., no. xxxx, 2021, doi: 10.1016/j.jksuci.2021.04.004.

[18] J. M. Sanchez-Gomez, M. A. Vega-Rodríguez, and C. J. Pérez, “Sentiment-oriented query-focused text summarization addressed with a multi-objective optimization approach,” Appl. Soft Comput., vol. 113, 2021, doi: 10.1016/j.asoc.2021.107915.

[19] I. Tanfouri, G. Tlik, and F. Jarray, “An automatic arabic text summarization system based on genetic algorithms,” Procedia CIRP, vol. 189, pp. 195–202, 2021, doi: 10.1016/j.procs.2021.05.083.

[20] Y. Du, Q. Li, L. Wang, and Y. He, “Biomedical-domain pre-trained language model for extractive summarization,” Knowledge-Based Syst., vol. 199, p. 105964, 2020, doi: 10.1016/j.knosys.2020.105964.

[21] T. T, M. Bohra, P. Dadure, and P. Pakray, “Comparative analysis of T5 model for abstractive text summarization on different datasets,” SSRN Electron. J., 2022, doi: 10.2139/ssrn.4096413.

[22] D. Suleiman and A. Awajan, “Deep Learning Based Abstractive Text Summarization: Approaches, Datasets, Evaluation Measures, and Challenges,” Math. Probl. Eng., vol. 2020, 2020, doi: 10.1155/2020/9365340.

[23] S. Maples, “The ROUGE-AR: A Proposed Extension to the ROUGE Evaluation Metric for Abstractive Text Summarization,” Symb. Syst. Dep., 2017, [Online]. Available: https://web.stanford.edu/class/cs224n/reports/2761938.pdf

[24] C. Irimia, F. Harbuzariu, I. Hazi, and A. Iftene, “Official Document Identification and Data Extraction using Templates and OCR,” Procedia Comput. Sci., vol. 207, pp. 1571–1580, 2022, doi: https://doi.org/10.1016/j.procs.2022.09.214.

[25] M. Junker and R. Hoch, “Evaluating ocr and non-ocr text representations for learning document classifiers,” in Proceedings of the Fourth International Conference on Document Analysis and Recognition, 1997, pp. 1060--1066.

[26] S. La Manna, A. M. Colia, and A. Sperduti, “Optical font recognition for multi-font OCR and document processing,” in Proceedings. Tenth International Workshop on Database and Expert Systems Applications. DEXA 99, 1999, pp. 549--553.

[27] W. Bieniecki, S. Grabowski, and W. Rozenberg, “Image Preprocessing for Improving OCR Accuracy,” in 2007 International Conference on Perspective Technologies and Methods in MEMS Design, 2007, pp. 75–80. doi: 10.1109/MEMSTECH.2007.4283429.

[28] F. LeBourgeois, “Robust multifont OCR system from gray level images,” in Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, 1997, pp. 1–5.

[29] C. Yao, X. Bai, and W. Liu, “A unified framework for multioriented text detection and recognition,” IEEE Trans. Image Process., vol. 23, no. 11, pp. 4737–4749, 2014.

[30] H. Takahashi, N. Itoh, T. Amano, and A. Yamashita, “A spelling correction method and its application to an OCR system,” Pattern Recognit., vol. 23, no. 3–4, pp. 363–377, Jan. 1990, doi: 10.1016/0031-3203(90)90023-E.

[31] C. C. Abalorio and M. Cerna, “Course Evaluation Generator (Ceg): An Automated Academic Advising System with Optical Character Recognition,” Int. J. Technol. Eng. Stud., vol. 4, no. 5, pp. 189–196, 2018, doi: 10.20469/ijtes.4.10003-5.

[32] D. Gifu, “AI-backed OCR in Healthcare,” Procedia Comput. Sci., vol. 207, pp. 1134–1143, 2022, doi: https://doi.org/10.1016/j.procs.2022.09.169.

[33] Hubert, P. Phoenix, R. Sudaryono, and D. Suhartono, “Classifying Promotion Images Using Optical Character Recognition and Naïve Bayes Classifier,” Procedia Comput. Sci., vol. 179, no. 2020, pp. 498–506, 2021, doi: 10.1016/j.procs.2021.01.033.

[34] C. Adjetey and K. S. Adu-Manu, “Content-based Image Retrieval using Tesseract OCR Engine and Levenshtein Algorithm,” Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 7, pp. 666–675, 2021, doi: 10.14569/IJACSA.2021.0120776.

[35] H. P. Luhn, “The Automatic Creation of Literature Abstracts,” IBM J. Res. Dev., vol. 2, no. 2, pp. 159–165, 2010, doi: 10.1147/rd.22.0159.

[36] V. A. Pitogo and C. D. L. Ramos, “Social Media Enabled E-Participation: A Lexicon-Based Sentiment Analysis Using Unsupervised Machine Learning,” in Proceedings of the 13th International Conference on Theory and Practice of Electronic Governance, 2020, pp. 518–528. doi: 10.1145/3428502.3428581.

[37] S. Kadagadkai, M. Patil, A. Nagathan, A. Harish, and A. MV, “Summarization tool for multimedia data,” Glob. Transitions Proc., vol. 3, no. 1, pp. 2–7, 2022, doi: 10.1016/j.gltp.2022.04.001.

[38] C. C. Abalorio, R. P. Medina, A. M. Sison, and G. A. Dalaorao, “Extended Max-Occurrence with Normalized Non-Occurrence as MONO Term Weighting Modification to Improve Text Classification,” Int. J. Adv. Comput. Sci. Appl., vol. 13, no. 4, pp. 91–97, 2022, doi: 10.14569/IJACSA.2022.0130411.

[39] C. C. Abalorio, A. M. Sison, R. P. Medina, and G. A. Dalaorao, “Applying EMONO Variants to Multi-Class Sentiment Analysis for Short-Distance Inter-Class Frequency of Term,” vol. 71, no. 4, pp. 1938–1947, 2022.