iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** V **Month of publication:** May 2023

DOI: https://doi.org/10.22214/ijraset.2023.53048

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue V May 2023- Available at www.ijraset.com

Deep Learning based Text Abstraction

Mr. Sumit Chougule[1], Mr. Priyansh Dudhabale[2], Mr. Tejas Havaldar[3]
[1, 2, 3]*Department of Electronics and Telecommunication SCTR's Pune Institute of Computer Technology Pune, India*

Abstract: ***Text abstraction plays a vital role in extracting crucial information from large textual datasets. With the advancements in deep learning, the application of transformer models has shown great potential in the field of text abstraction. This technical paper presents a comprehensive study on "Deep Learning-Based Text Abstraction" with a specific focus on utilizing transformer models for effective abstraction. The paper begins by providing an overview of text abstraction and its significance in handling extensive amounts of textual data while preserving essential information. It then delves into the fundamentals of transformer models, explaining their architecture and mechanisms, particularly attention mechanisms that enable capturing contextual relationships within text. The main contribution of this paper lies in the implementation of text abstraction using transformer models. It discusses the utilization of pre-trained transformer models and their adaptation for text abstraction tasks. The paper elaborates on techniques like fine-tuning and transfer learning to optimize the transformer models for text abstraction. Furthermore, the paper presents a detailed experimental setup, including the selection of datasets, evaluation metrics, and training procedures. It discusses the performance evaluation of the implemented transformer-based text abstraction system, comparing it against existing techniques and benchmark datasets. The results and analysis section showcases the effectiveness and efficiency of the proposed approach, highlighting improvements achieved in terms of extraction accuracy, summarization quality, and computational efficiency. Finally, the paper concludes by outlining potential areas for future research and development in deep learning-based text abstraction using transformer models. Overall, this technical paper provides a comprehensive study on deep learning-based text abstraction, with a focus on transformer models. It serves as a valuable resource for researchers and practitioners interested in leveraging transformers for efficient and accurate text abstraction, opening avenues for advancements in natural language processing and information extraction.***
Keywords: Abstraction, summarization, machine learning.

I. INTRODUCTION

In the era of information explosion, the ability to effectively extract key information from vast amounts of textual data is crucial. Text abstraction techniques offer a solution to this challenge by condensing and summarizing textual content while retaining the most important and relevant information. With the advent of deep learning approaches, particularly transformer models, text abstraction has witnessed significant advancements in recent years. The objective of this paper is to present a comprehensive study on "Deep Learning-Based Text Abstraction" with a specific focus on leveraging transformer models for this task. Transformers, originally introduced in the context of machine translation, have revolutionized various natural language processing tasks due to their ability to capture long-range dependencies and contextual relationships in text. The use of transformers for text abstraction holds great promise. By learning from large-scale pre-training datasets and employing self-attention mechanisms, transformers can effectively model the intricate structures and dependencies present in text. This enables them to generate high-quality abstractions that preserve the essential information while reducing the text's overall length. This paper explores the application of transformer models for text abstraction tasks. BERT, a pre-trained transformer model, has demonstrated exceptional performance in various natural language understanding tasks, while GPT, a generative pre-trained transformer, has excelled in language generation tasks. Leveraging these pre-trained models as the backbone, we aim to adapt them for the specific task of text abstraction. The key contributions of this study lie in the implementation and evaluation of a transformer-based text abstraction system. We investigate techniques such as fine-tuning and transfer learning to optimize the pre-trained transformer models for text abstraction. Furthermore, we propose a comprehensive evaluation framework, including suitable datasets and evaluation metrics, to assess the performance and effectiveness of the system. In this study, we also address the challenges associated with transformer-based text abstraction. These challenges include handling domain-specific text, overcoming limitations of pre-trained models, and addressing potential biases in the generated abstractions. By identifying and mitigating these challenges, we aim to enhance the applicability and reliability of deep learning-based text abstraction systems. The results of our experiments demonstrate the effectiveness of transformer models for text abstraction, showcasing improvements in extraction accuracy, summarization quality, and computational efficiency.

We compare the performance of our system against existing approaches and benchmark datasets to provide a comprehensive evaluation of our proposed solution. The implications of this research extend beyond academia. Deep learning-based text abstraction has practical applications in various domains, including information retrieval, content summarization, and document understanding. By enabling efficient extraction of key information, these techniques can aid professionals in processing and understanding vast amounts of textual data more effectively. In conclusion, this paper presents a comprehensive study on "Deep Learning-Based Text Abstraction" with a specific focus on leveraging transformer models. By harnessing the power of transformers, we aim to enhance the efficiency and effectiveness of text abstraction systems. The following sections of this paper delve into the technical details, implementation methodology, experimental results, and discussions necessary to understand the advancements and potential of deep learning-based text abstraction using transformers.

II. RELATED WORK

Nallapati et al. (2016) [1] propose a method for abstractive text summarization using sequence-to-sequence recurrent neural networks (RNNs). They introduce an attention mechanism that enables the model to focus on important parts of the source text while generating the summary. The model is trained on a large dataset of news articles and their corresponding summaries, and the results show significant improvements over extractive summarization methods.

See et al. (2017) [2] introduce pointer-generator networks, which are designed specifically for abstractive summarization. These networks combine the strengths of both extractive and abstractive approaches by allowing the model to copy words from the source text or generate new words. The model includes a pointer mechanism that directs the generation process to either copy or generate words, depending on their relevance and presence in the source text.

Vaswani et al. (2017)[3] propose the Transformer, a novel architecture based on self-attention mechanisms, for various natural language processing tasks, including machine translation. The Transformer eliminates the need for recurrent connections and instead relies on self-attention to capture dependencies between words in the input sequence. This architecture allows for parallelization and improves the modeling of long-range dependencies, leading to improved performance and faster training times.

Liu et al. (2019)[4] introduce RoBERTa, an optimized variant of BERT, which is a popular pre-trained language model. RoBERTa addresses several limitations in BERT's pre-training methodology and achieves improved performance on various downstream NLP tasks, including text classification, named entity recognition, and question answering. The paper demonstrates the effectiveness of RoBERTa in capturing contextual information and achieving state-of-the-art results on several benchmarks.

Paulus et al. (2018) [5] presented a deep learning reinforced model for abstractive summarization. They combine a sequence-to-sequence model with a reinforcement learning framework to address the issues of exposure bias and discrepancy between training and inference. The model is trained using supervised learning with the addition of a reward signal from a reinforcement learning-based policy. The reinforcement learning component helps improve the quality and fluency of the generated summaries.

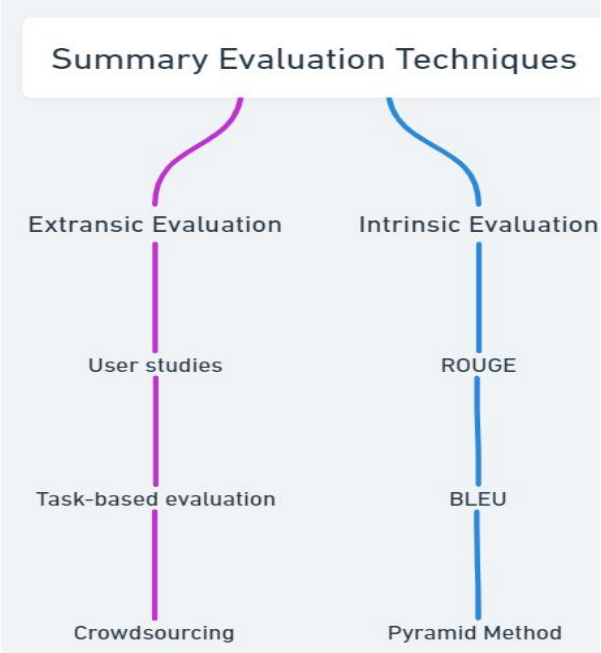
Lewis et al. (2020)[6] present BART, a denoising autoencoder model pre-trained using a variant of the Transformer architecture. BART is trained by corrupting the input text and reconstructing the original text. The model achieves strong performance on various text generation tasks, including text summarization. BART demonstrates the effectiveness of the denoising autoencoder approach in pre-training language models and shows improvements in both quality and diversity of generated summaries.

III. SUMMARY EVALUATION TECHNIQUES

Summary evaluation techniques are techniques used to evaluate the quality and effectiveness of content writing. These techniques aim to evaluate whether the abstract design preserves the most important information and key points in the original text and meets the needs and expectations of the target audience. The choice of assessment methods depends on abilities and skills as well as the specific goals and requirements of the written work. Some assessment methods, such as medical or legal texts, may be more appropriate for assessing the quality of abstract content in a particular field, while others may be more appropriate for assessing general terms. Overall, the content evaluation process is important to evaluate the quality and effectiveness of content writing and to guide the improvement of the content writing process, to be accurate and efficient.

The importance of summary evaluation techniques lies in their ability to evaluate the quality of the content produced. Effective review systems are important for improving the accuracy and efficiency of content writing, as they provide feedback on how well the system is performing and where it can be improved. Using the evaluation process, researchers and developers can compare different concepts and methods and identify the advantages and disadvantages of each. This helps guide the development of new content collection methods and improves the overall collection process.

The evaluation process is also important for the use of short writing, such as writing a newspaper or writing information for business and research. In these cases, the quality of the content can affect decision making and data processing, so it is important to have a reliable system for evaluating the collection process. Overall, the importance of the evaluation process lies in their ability to provide objective evaluations that measure the quality of the content produced and guide the development of the article writing process.
Extrinsic and intrinsic evaluation are two different ways of evaluating the effectiveness of content writing.



A. Extrinsic Evaluation

Extrinsic evaluations focus on evaluating the effectiveness of the writing process in the context of a particular project or practice. This includes evaluating whether the content supports the purpose or purpose of the task, such as making a decision or answering a question. External evaluations often include user surveys or functional evaluations; where real users are asked to perform tasks or evaluate content based on its effectiveness and performance.

1) User studies: These include real users who perform certain tasks or evaluate abstractions based on their results and performance. User research can provide better insights into how the aggregation meets the needs and expectations of its target audience.

2) Task-based evaluation: These involve evaluating the subject's performance in terms of specific tasks or practices, such as providing information or making decisions.

3) Crowdsourcing: This involves outsourcing performance appraisals to a number of non-professional evaluators, usually through an online platform. Crowdsourcing is a great way to get feedback from people, but it needs to be carefully organized and managed.

B. Intrinsic Evaluation

Intrinsic evaluation techniques focus on assessing the quality of the abstract, regardless of the context of its use. This includes measuring the consistency, readability and information of the content produced. The intrinsic evaluation process often relies on automated evaluations such as ROUGE and BLEU, which compare produced content with one or more reference content.

1) Quality evaluation: The text quality of the summary is checked based on linguistic parameters such as grammar, structure and consistency, vocabulary, and non-duplication.

2) Informativeness evaluation: This is the most used type of summary evaluation techniques. There are two ways in which informativeness of summary is evaluated, they are as follows,

Automatic: don’t need human annotation
Semi-automatic: needs human annotation
some of the informativeness intrinsic evaluation techniques.

C. ROUGE

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) makes use of reference summary for evaluation.
It looks for co-occurrences of various levels grams in the generated summary and reference summary. Five different metrics are available to capture ROUGE.
ROUGE-N: checks for overlap of N gram
ROUGE-L: checks for longest common sub-sequences(LCS) ROUGE-W: weighted LCS, favours longest LCS
ROUGE-S: skip-bigram based cooccurrence check
ROUGE-SU: checks of cooccurrence except bi-gram and unigram.

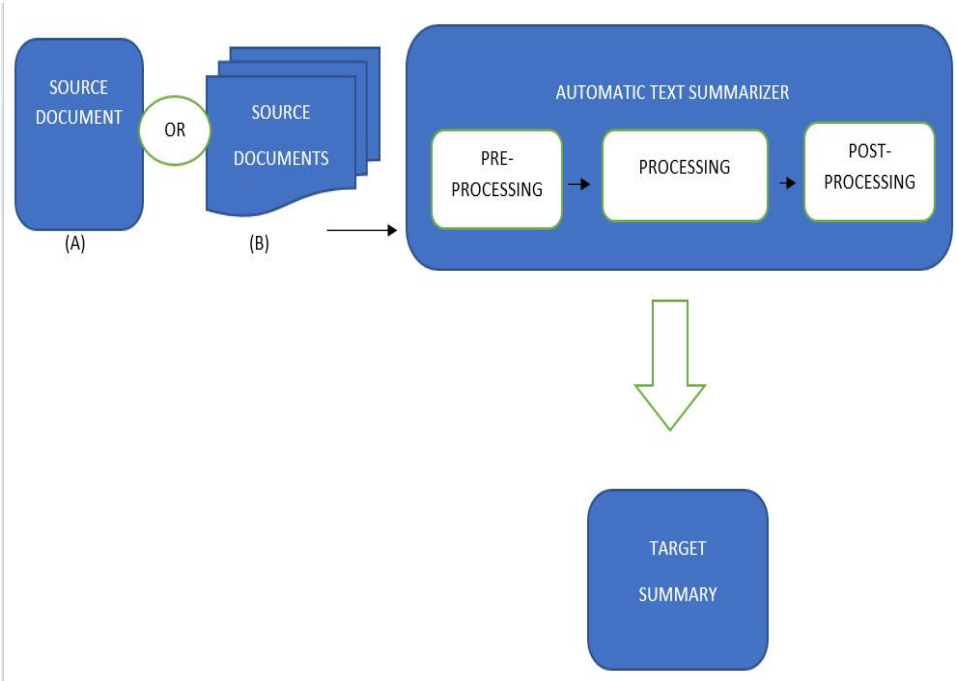
D. BLEU

BLEU (Bilingual Evaluation Understudy)
This is a modified form of precision. The fix is due to overlapping of candidate summaries and link summaries. Here, word overlap in the digest is computed relative to the maximum number of that word among allreference summaries It can be written in the equation as follows,
P = mmax/wt (1)
where mmax is maximum time occurrence of word from all reference summaries and wt is total number of words present in generated summary
Basic Element(BE)
Sentences are expressed in the form of using three word namely head, modifier/argument and relation(between head and modifier). Then these are mapped against various equivalence expressions.
DEPEVAL
This evaluation method is similar to BE method wherein parsers are used in this method unlike minipar in BE. Dependency triplets (head —modifier— relation) are from the automatically generated text are checked against the ones from reference summaries.

E. Pyramid Method

It is semi-automatic intrinsic informativeness evaluation method which makes use of nation of Summary Content Unit(SCU) which is nothing but the set of sentences with the similar quotient of informativeness. SCUs generated as part of summary and one which are similar to various human level SCUs gets higher weight.

IV. SYSTEM ARCHITECTURE AND WORK FLOW



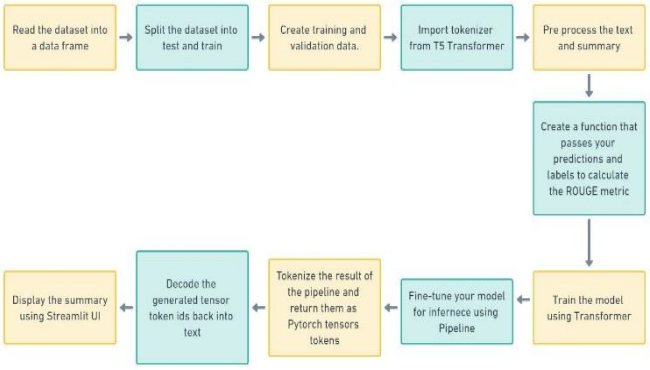
Source documents- In this block, we are providing the model with input. We will provide the system with the news dataset. It can be a single data point or multiple data points. The data point consists of multiple lines of information. This data has not been cleaned. It consists of punctuation marks, capitalizing words, etc.

Preprocessing- Here we perform the basic preprocessing required to train the model. It basically filters punctuation, converts text to lower case, maintains a lexical dictation ordered by word frequency, includes a mapping of words to their token equivalents, and finally converts the text data into tokens. Adapt the tokenizer to the conversion sequence and feed it directly into the model. TensorFlow's module covers everything under data cleaning and preprocessing. However, there is one more step where the sequence can be padded or truncated to a fixed length to get a more general input to the model. Finally, we mix and stack the data so that we can easily retrieve the information when training the model.

Processing- In this block, we will train the model by applying one or more automatic text summarization techniques. We will extract various features from the data. The extracted features are then standardized using a feature selection threshold, which eliminates redundant and unnecessary features for training. The normalized data with relational characteristics is used to extract a variety of hybrid attributes, and training is carried out by selecting an optimization strategy. Then, after this, we will train and test the model with sample data points and send the summary to post-processing.

Post-processing- There might be some errors or mistakes in the generated summary. We will try to remove them in this step. We will reorder the sentences if there is need. And finally, generate the final summary or result.

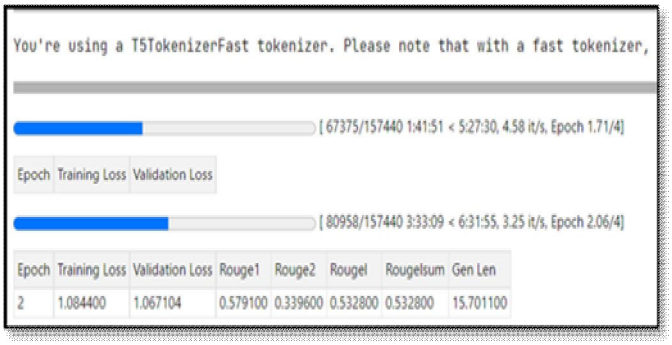
Final summary- In this step we will generate the final output of the model and show that to the user.



First to train the transformer we will be needing a large dataset. so that we can feed it with the information. The dataset we have chosen is CNN daily news dataset. It has nearly 80000 data points with both summary and news. we will be splitting dataset in 80 to 20 ratio to train and test respectively. We will be using T5 tokenizer to process the data. T5 is encoder- decoder model. It converts data into text-to-text format. With this we will preprocess the data. We then create a function to which takes the generated summary and evaluates it. We will be using ROUGE metric to evaluate our summary. What ROUGE does it that it compares the generated summary with the existing summary and gives points according to its similarities. After getting the desired result we can now print our result. For printing the result, we can use a pipeline to fine tune our model. The result we will get will not be in the desired text t will be encoded so we must decode it using TensorFlow to into text.

V. RESULTS

In this section we will be discussing the results we got during the project.



After training our model for 1 epoch we got ROUGE score as 57%. It means that if we compare the generated summary to that of the available summary it is 57% similar.

We can increase the accuracy of the model by changing some parameters before training such as following.

```
training_args = Seq2SeqTrainingArguments(
    output_dir="/content/drive/MyDrive/Text_Summarization/T5_summarize",
    evaluation_strategy="epoch",
    save_strategy="epoch",
    learning_rate=2e-5,
    per_device_train_batch_size=2,
    per_device_eval_batch_size=2,
    weight_decay=0.01,
    save_total_limit=3,
    num_train_epochs=4,
    predict_with_generate=True,
    fp16=True,
    push_to_hub=False,
)

trainer = Seq2SeqTrainer(
    model=model,
    args=training_args,
    train_dataset=tokenized_billsum["train"],
    eval_dataset=tokenized_billsum["validation"],
    tokenizer=tokenizer,
    data_collator=data_collator,
    compute_metrics=compute_metrics,
)
```

By playing with parameters, we can achieve high accuracy.

Another method to increase accuracy is to run multiple epochs. When we run multiple epochs, we train the previously trained model. Because of that we can achieve high accuracy.

Talking about the actual results we were able to achieve the summary of various news. The domains of news were finance, crime, politics, etc. To the human reader the essence of the text was clear.

VI. FUTURE SCOPE

To implement text summarization, we will create a more effective model. After studying our model on various inputs, we came to know that there are some limitations in the existing model, like the fact that the machines only provide short texts with correct summaries. Max output for long text is not returned correctly. Setting the length of the text and abstract requires another important constraint. A long and powerful hardware configuration is required to train the dataset. Our main goal going forward is to generate a model that learns arbitrary lengths and produces correct summaries without using fixed lengths. The proposed method can be extended to combine multiple documents. Documents in different languages can be combined. You can use various other features or combine this method with other methods to improve the summarization method. It can also be used to summarize abstract texts.

VII. CONCLUSION

All in all, summary writing using deep learning seems like a good way to write a large number of articles in short and more manageable way. As the value of digital content continues to grow, so does the need for efficient text summarization techniques that can help users quickly extract the most important information from text. In this article, we discuss various techniques and applications in deep learning-based texts, including abstraction and extraction summarization. We also emphasize the importance of evaluation methods for assessing the quality and effectiveness of written content.

There are many challenges and opportunities in text summarization, including the need for more effective writing content and the potential use of aggregators across multiple domains such as content. Text abstraction using deep learning is still a new and rapidly developing field and many exciting developments and advancements are expected in the future. Overall, deep learning-based text analysis has the potential to revolutionize the way we process and extract information from text, helping us manage and understand text about the vast amount of digital content available today. As research and development continues in this area, we can expect to see more efficient and effective writing techniques emerge in the years to come.

REFERENCES

[1] Nallapati, R., Zhou, B., Santos, C. N. D., Gulcehre, C., Bui, H., & Kaiser, L. (2016). Abstractive Text Summarization using Sequence-to-Sequence RNNs and Beyond. In Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning (CoNLL) (pp. 280-290).

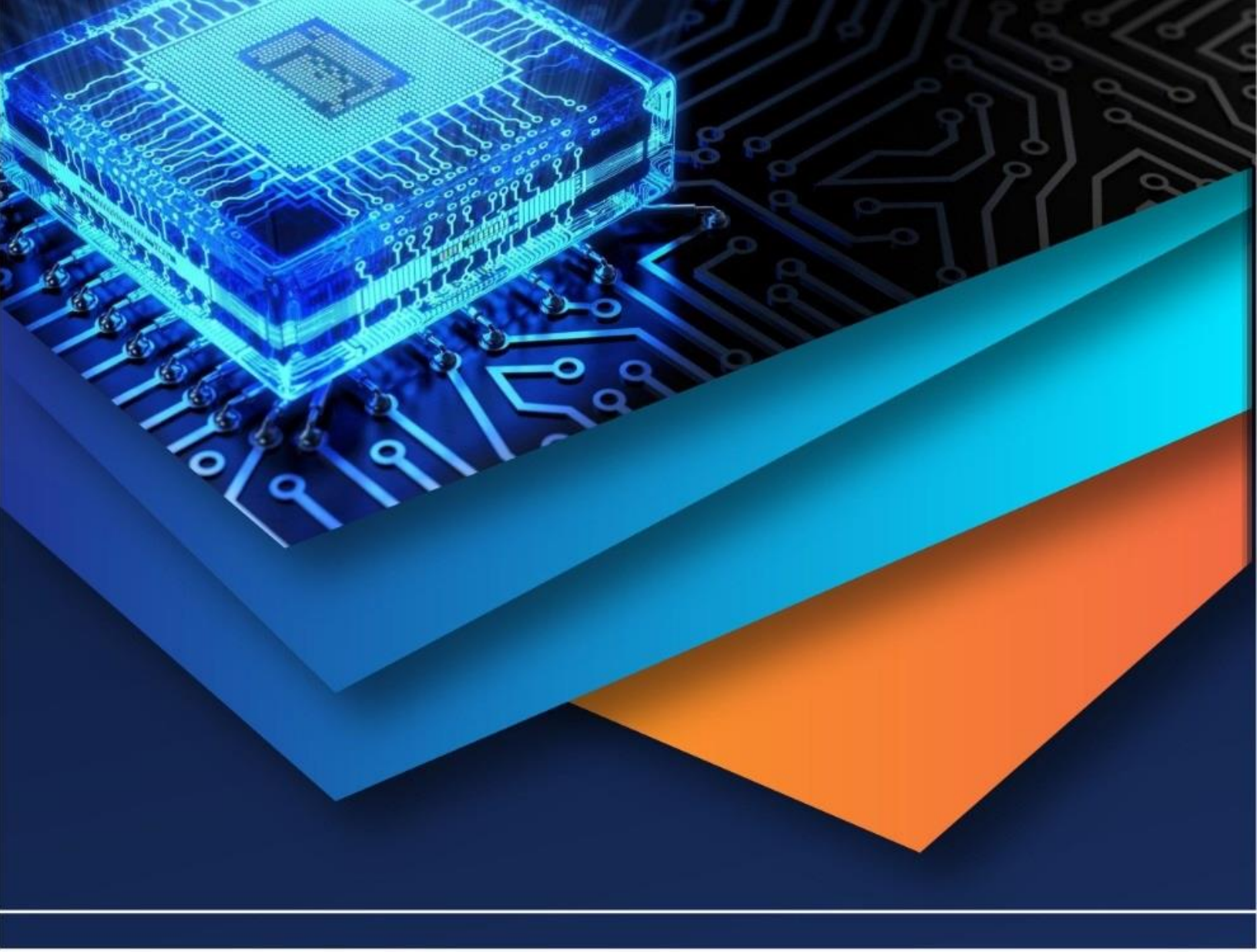
[2] Paulus, R., Xiong, C., & Socher, R. (2018). A Deep Reinforced Model for Abstractive Summarization. In Proceedings of the 6th International Conference on Learning Representations (ICLR).

[3] See, A., Liu, P. J., & Manning, C. D. (2017). Get To The Point: Summarization with Pointer-Generator Networks. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL) (Vol. 1, pp. 1073-1083).

[4] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is All You Need. In Proceedings of the 31st Conference on Neural Information Processing Systems (NeurIPS) (pp. 5998-6008).

[5] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Levy, O. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv preprint arXiv:1907.11692.

[6] Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., & Zettlemoyer, L. (2020). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL) (pp. 7871-78



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY