Online ISSN 3007-3197
Print ISSN 3007-3189
http://amresearchreview.com/index.php/Journal/about

Annual Methodological Archive Research Review
http://amresearchreview.com/index.php/Journal/about
Volume 3, Issue 5 (2025)

Automatic Abstractive Summarization of Text: Harnessing the Power of Large Language Models and Deep Learning

[1]Ali Raza, [2*]Muhammad Waseem Iqbal, [3]Nazim Hussain, [4]Khalid Hamid, [5]Hannan Liaqat, [6]Muhammad Akram Mujahid, [7]Sehar Gul

Article Details

Keywords: Automatic Summarization; NLP; T5; BART; GPT-2; EGPT-2

Ali Raza
Department of Computer Science and Information Technology, Superior University Lahore, Punjab, Pakistan.
oarazarila@gmail.com
Muhammad Waseem Iqbal*
Department of Software Engineering, Superior University Lahore, Punjab, Pakistan. Corresponding Author Email: waseem.iqbal@superior.edu.pk
Nazim Hussain
Lecturer Minhaj University Lahore, Lahore, Pakistan. nazimhussain.cs@mul.edu.pk
Khalid Hamid
Department of Computer Science & IT Superior University Lahore, Punjab, Pakistan.
Khalid6140@gmail.com
Hannan Liaqat
Department of Information Science, DSNT, University of Education, Lahore, 54000, Pakistan. hannanliaqat@hotmail.com
Muhammad Akram Mujahid
Department of Information Science, DSNT, University of Education, Lahore, 54000, Pakistan. akram.mujahid@ue.edu.pk
Sehar Gul
Department of Computer Science, Sukkur IBA University, Sindh, Pakistan. sehar.gul@iba-suk.edu.pk

ABSTRACT

In today's world, much information is available online and offline. There are hundreds of articles on a single topic with a lot of data. It isn't easy to extract helpful information manually. Automatic text summarization systems have been developed to solve this problem. Text summarization takes valuable information from large documents and puts it in short summaries. There are two ways to generate summaries: extractive summarization and abstractive summarization. Only relevant sentences are extracted from the original document using the extractive technique. Abstractive summarization involves interpreting the original text before generating the summary. A lot of research on extractive summarization. However, analysis in abstractive summary in the Urdu language has not been studied well so far. Urdu is a dynamic language in terms of literary sources and requires serious research efforts to generate abstractive summaries. So, we propose an abstractive summarization method for Urdu text using the Urdu Fake News dataset and pre-trained models. For analysis, Urdu Fake News dataset is used containing text data for summarization and to serve this purpose we provide four summarization systems: one based on BART, another on T5, the third on GPT-2, and a fourth on EGPT-2. We employ a pre-trained model designed explicitly for Urdu to perform the summarization task. This study intends to enhance Urdu text summarization by identifying inefficient heads and then removing them entirely from the model. We evaluated our suggested abstractive summarization model using the Rouge Score and found that it improved accuracy and produced more natural, cohesive summaries.

INTRODUCTION

The growing volume of online text, such as news and scholarly articles, has increased the demand for automatic summarization over the past two decades [1]. With vast amounts of text requiring processing, machines are essential for summarizing content efficiently. While text summarization has been extensively studied for languages like English, Urdu has received less attention due to its grammatical and morphological complexity, posing significant challenges for Natural Language Processing (NLP) [2].

Text summarization reduces lengthy documents into concise summaries while preserving key information and meaning [3], saving time in analyzing documents like research papers. It is categorized into extractive and abstractive methods. Urdu, an Indo-Aryan language spoken by over 100 million people globally, serves as the national language of Pakistan [5-8]. Influenced by Arabic, Persian, and Sanskrit, Urdu's 38-character alphabet and complex syntax make summarization challenging [4]. Its unique vocabulary and morphology further complicate NLP tasks. Current Urdu summarization systems often combine extractive techniques (e.g., word frequency, TF-IDF) with abstractive approaches, using models like BERT to refine outputs [10]. However, abstractive summarization remains particularly challenging [11], highlighting the need for dedicated systems tailored to Urdu [12].

This paper proposes an abstractive summarization method for Urdu using the Urdu Fake News dataset and pre-trained models. We implemented three systems based on BART, T5, and GPT-2, employing a pre-trained Urdu-specific model for summarization. Our approach identifies and removes inefficient attention heads to enhance performance. Evaluated using ROUGE scores, the model demonstrated improved accuracy and produced more cohesive summaries. Limitations include challenges with ambiguous or informal language and the scarcity of annotated Urdu corpora, which may limit generalizability across topics or dialects. Despite these constraints, this research contributes to the sparse literature on Urdu NLP, addressing a critical gap in automated summarization. It lays the groundwork for advanced applications like real-time summarization tools, chatbots, and educational resources. Future work will focus on expanding datasets, exploring diverse models, and testing new architectures to further improve Urdu abstractive summarization [45].

The subsequent sections of this paper are organized as follows: Section 2 reviews existing literature on abstractive text summarization, emphasizing foundational methodologies and gaps in Urdu-language research. Section 3 outlines the architecture and workflow of the proposed Urdu abstractive summarization system, including data preprocessing, model design, and training protocols. Section 4 discusses the evaluation framework, detailing performance metrics and validation methodologies. Section 5 presents implementation tools, experimental results, and comparative analyses through quantitative tables and graphical visualizations. Finally, Section 6 concludes the study by synthesizing key findings, addressing limitations, and proposing future directions for advancing Urdu NLP summarization systems.

LITERATURE REVIEW

In this section we review technical background and state-of-the-art advancements in automatic summarization. In [23], a project categorized social media posts into topics like 'politics' and 'sports' using multilingual BERT on Arabic datasets from news networks. Data was collected from Twitter, Facebook, and other platforms, covering both Modern Standard and colloquial Arabic. The model was trained on various datasets, including social media posts, web articles, and Twitter data, achieving high accuracy.

Researchers have developed techniques for abstractive text summarization (ATS) using key corpora. In [24], a convolutional encoder with attention was proposed for sentence-level summarization, tested on DUC-2004 and Gigaword, achieving ROUGE-1 = 31.0, ROUGE-2 = 12.6, and ROUGE-L = 28.3 on Gigaword. In [25], convolutional RNN and LSTM encoder-decoder models were applied to Gigaword and DUC-2004. A recurrent attention model outperformed LSTM on Gigaword, with ROUGE-1 = 33.7, ROUGE-2 = 15.9, and ROUGE-L = 31.1. In [26], a copy mechanism in sequence-to-sequence RNN addressed out-of-vocabulary issues on LCSTS, replicating input segments in outputs. Tested at word and character levels, the best results on the word level were ROUGE-1 = 35.0, ROUGE-2 = 22.3, and ROUGE-L = 32.0.

In [27], a pointer-generator coverage model avoided repetition by preserving attention history, applied to CNN/Daily Mail. The model achieved ROUGE-1 = 39.5, ROUGE-2 = 17.2, and ROUGE-L = 36.3. In [28], a content selection method with attention models improved word choice for summaries. Applied to CNN/Daily Mail and NYT, the bottom-up approach achieved ROUGE-1 = 41.2, ROUGE-2 = 18.6, and ROUGE-L = 38.3. In [29], a global encoding model eliminated recurrence, retaining semantically relevant information. Applied to LCSTS and Gigaword, it achieved ROUGE-1 = 39.4, ROUGE-2 = 26.9, and ROUGE-L = 36.5 on LCSTS.

In [30], the authors introduced adding decoder input words into the attention mechanism for the first time when calculating attention vectors to improve abstractive summaries. The attention mechanism proposed in this work also incorporates semantic and contextual similarities. The authors applied the proposed attention mechanism along with other baseline techniques on the CNN/Daily Mail [4] and Gigaword [12] corpora. The best results were obtained from the proposed attention mechanism on Gigaword corpus with ROUGE-1 = 38.2, ROUGE-2 = 16.4 and ROUGE-L = 36.0.

In [31], the authors address factual consistency in abstractive summaries by filtering training data and applying transfer learning models like BERT and Pegasus on Newsroom, Xsum, and CNN/Daily Mail corpora. The best ROUGE scores were achieved using BERT on Xsum: ROUGE-1 = 45.6, ROUGE-2 = 22.5, ROUGE-L = 37.2. In [32], the authors improve summary quality by decomposing the decoder to extract key sentences and refining abstractive summaries with language models. Using reinforcement learning on the CNN/Daily Mail corpus, they achieved ROUGE-1 = 40.1, ROUGE-2 = 17.3, and ROUGE-L = 37.5.

To address the need for efficient summarization of multimodal social media content in low-resource languages, [39] introduced UrduMASD, the first multimodal abstractive summarization dataset for Urdu, featuring 15,374 videos with audio, titles, transcripts, and human-written summaries. Quality assessments using metrics like Abstractive and Compression showed superior performance over existing datasets. Baseline experiments demonstrated a 2.6% ROUGE score improvement when incorporating visual features, highlighting the importance of multimodal inputs for advancing Urdu NLP. In [44] [40], an encoder-decoder architecture was proposed for text generation tasks. The encoder converts input sequences into context vectors, while the decoder generates outputs using embedding and LSTM layers. A dense layer with SoftMax produces vocabulary-based probabilities. For Urdu summarization, the authors combined SVM classifiers with TF-IDF scoring to rank sentences, demonstrating the effectiveness of integrating statistical and neural methods.

In [41], the authors addressed abstractive summarization for Roman Urdu (RU), a low-resource language used in social media. They created a Roman Urdu corpus by transliterating data from two Urdu summarization datasets. Baseline tests using BERT and T5 transformer models showed that T5 outperformed BERT in generating abstractive summaries. The study highlights the potential of transformer models for Roman Urdu summarization and calls for further research in this areaThis template, modified in MS Word 2007 and saved as a "Word 97-2003 Document" for the PC, provides authors with most of the formatting specifications needed for preparing electronic versions of their papers.

PROPOSED METHODOLOGY

In this section detailed implementations of proposed model is discussed.

TRANSFORMER MODELS

The Transformer architecture, initially designed for natural language processing (NLP) tasks [24], has been successfully applied to domains like image generation and reinforcement learning. It leverages self-attention to extract significant features from input sequences by focusing on different parts of the sequence. The self-attention mechanism computes a weighted sum of sequence elements based on their similarity, with weights determined by this similarity [43] [25]. The Transformer consists of encoders and decoders, each comprising multiple self-attention layers and feedforward neural networks [42] [33]. The encoder generates semantic representations of the input sequence, while the decoder predicts subsequent elements in the sequence based on previously generated ones. During decoding, an attention mechanism connects the encoder and decoder, allowing the decoder to focus on specific parts of the encoded input by computing a weighted sum of encoded representations based on similarity, as illustrated in Fig. 1. Transformers have revolutionized NLP, excelling in tasks like language modeling, machine translation, and text categorization [26]

EFFICIENT GPT-2 SUMMARIZER MODEL

Our objective is to enhance the precision of abstractive Urdu text summarization using the Efficient GPT-2 Summarizer model. In natural language processing (NLP), Efficient GPT-2 is considered one of the most reliable pre-trained transformer models due to its adaptability and robustness. By applying this model to cleaned and preprocessed data, we aim to generate more accurate and coherent summaries, improving the overall quality of the summarization process.

We focus on identifying ineffective attention heads to enhance the Efficient GPT-2 Summarizer model's performance in abstractive Urdu text summarization. These attention heads play a critical role in extracting key information during the summarization process. By analyzing their attention weights and distribution, we can determine whether they contribute meaningfully to the overall representation or are largely redundant. This analysis helps us prioritize or deprioritize specific attention heads, improving the quality of the generated summaries. We analyze the attention heads and then provide a method for deleting the ineffective ones from the Efficient GPT-2 Summarizer model. To improve the model's efficiency, we may get rid of the attention heads that do not substantially contribute to the summarizing work. In order to improve the quality of the generated summaries, the model's architecture is refined, and the remaining attention heads are optimized. Algorithm for EGPT is presented as:

PREPARING THE DATA: Gather an archive of Urdu text documents and their summaries for use in text analytics. Tokenize the text and encode the summaries as a first step in processing the data. This is an important stage since it sets the stage for subsequent processing of the data.

TRAINING FOR THE EGPT-2 SUMMARIZER MODEL: Use the cleaned-up Urdu text dataset to train an Efficient GPT-2 Summarizer model. The model is trained to provide abstract but cohesive summaries of the source material.

IMPORTANCE OF THE ATTENTION HEAD EVALUATION: Optimize Efficient GPT-2 Summarizer model for text summarizing based on its learned data. Compare the model's results with both sets of attention heads and conclude which is more beneficial. Evaluate the effect of eliminating each attention head using applicable assessment criteria like ROUGE [34] scores or other suitable metrics.

SET A THRESHOLD AND RANK ATTENTION HEADS: Set a threshold to determine the minimum importance required for attention heads to be retained. Rank the attention heads based on their importance, from most important to least.

REMOVE INEFFICIENT ATTENTION HEADS REMOVE: Attention heads that fall below the importance threshold. Adjust the model architecture and attention mechanisms to ensure the remaining attention heads adequately capture the relevant information for text

RETRAIN AND EVALUATE THE PRUNED MODEL: Retrain the updated Efficient GPT-2 Summarizer model using the preprocessed Urdu text dataset. Evaluate the pruned model's performance in generating abstractive summaries using appropriate evaluation metrics such as ROUGE scores or other relevant metrics. Evaluate how removing ineffective attention heads affected model performance by comparing the trimmed and unpruned versions

DATASET

The Urdu fake news dataset [35] is a collection of news articles in the Urdu language that have been labeled as either "real" or "fake" news. This dataset is available on Hugging Face, a popular platform for sharing and discovering datasets for natural language processing (NLP) tasks. The dataset contains 12,000 news articles, with 6,000 labeled as real news and 6,000 labeled as fake news. Detailed statistics of dataset is illustrated in Table.1. The dataset comprises news articles from six distinct categories: technology, education, business, sports, politics, and entertainment. The authentic news articles were gathered from various reputable mainstream news platforms in Pakistan, India, the UK, and the USA [36]. Some of the sources include BBC Urdu News [37], CNN Urdu [38], Express-News, Jung News, and several other trusted news websites.

TABLE I. STATICTICS OF DATASET

Category	Real News	Fake News	Total Articles	% of Overall Articles
Politics	1,800	1,800	3,600	30%
Business	1,200	1,200	2,400	20%
Technology	900	900	1,800	15%
Sports	900	900	1,800	15%
Education	600	600	1,200	10%
Entertainment	600	600	1,200	10%
Total	6,000	6,000	12,000	100%

DATA PREPROCESSING

Tokenization: Tokenization is the first step in preprocessing, which breaks the document into smaller paragraphs, sentences, and words. The text must then be cleared of stop words to prevent repetition. Next, stemming is performed to convert derived words back into their stems. The lemmatization of text is also an essential step in text preprocessing. To understand what a word means, you need to know its context as shown in Fig. 1. The Urdu language typically separates words with spaces, so this step may involve splitting the text at every space.

Stop word removal: Stop words are common words with little meaning that can be safely removed from texts without losing important information. As shown in Fig. 2(b).

SENTENCE SEGMENTATION: Sentence segmentation is the process of breaking down a text into its individual sentences. Many natural language processing applications, including sentiment analysis, text classification, and machine translation, depend on the segmentation of Urdu text. Usually, a period (.), a question mark (?), or an exclamation point (!) separates two Urdu sentences. However, it's vital to keep in mind that these punctuation marks can also be used within sentences (for instance, in acronyms and quoted text).

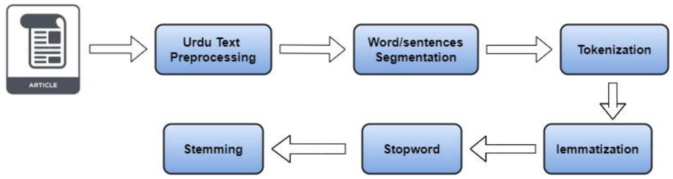


Fig. 1. URDU TEXT PREPROCESSING ASPECTS.

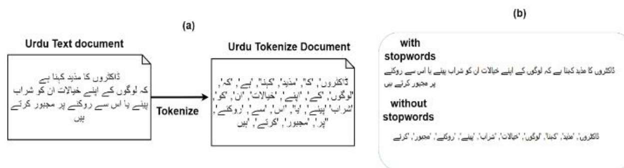


Fig. 2. (A) SENTENCE TOKENIZATION. (B) SENTENCE WITH AND WITHOUT STOP-WORDS

STEMMING: a Language Processing Method Called Stemming Breaks Down Words To Their Root Or Fundamental Form. As Shown In Fig. 3.

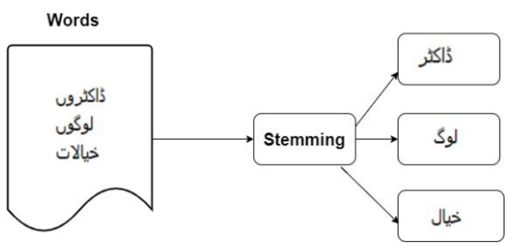


Fig. 3. stemming OF URDU TEXT.

LEMMATIZATION: The process of lemmatization reduces words to their most basic unit, known as the lemma. Lemmatization identifies the base word of a given word by removing inflectional or derivational suffixes.

ENVIRONMENTAL SETTINGS

Experiments took place within a system that used Ubuntu 20.04 LTS as its operating system and contained an Intel Core i7-9700K CPU operating at 3.6 GHz with 8 cores together with 32 GB of DDR4 RAM to provide sufficient capacity for deep learning operations. The project used an NVIDIA RTX 2080Ti GPU along with 11 GB of GDDR6 VRAM as the model accelerator alongside a 1 TB SSD for storing dataset files and model checkpoint information. Our project processed through Python 3.8 while using PyTorch 1.9.0 as the deep learning framework together with important NLP libraries Hugging Face Transformers and Datasets which operated with SentencePiece for multilingual tokenization. To maintain systems compatibility the project used Python's venv and Docker containerization to create separate development environments..

PERFORMANCE EVALUATION

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is a set of metrics used to evaluate the quality of generated summaries by comparing them to human-created "golden reference" summaries. It measures overlap using word pairs, sequences, and n-grams. Common ROUGE metrics include: ROUGE-N measures overlap of N-grams (e.g., unigrams, bigrams) between system and reference summaries. ROUGE-L captures the longest common word sequence using the Longest Common Subsequence (LCS) algorithm. ROUGE-S evaluates skip-bigrams, allowing gaps between words in a sequence. ROUGE-SU combines ROUGE-S and ROUGE-L as a weighted average. Most studies focus on ROUGE-1, ROUGE-2, and ROUGE-L to assess summary quality in terms of informativeness, coherence, and fluency [19]. In this research, we evaluate the improved Efficient GPT-2 Summarizer model for abstractive Urdu text summarization using these metrics. The model is rigorously tested on a dataset to determine its ability to generate informative and coherent summaries. Progress achieved by eliminating ineffective attention heads is measured against established benchmarks. Results from three summarizers are compared using ROUGE scores to quantify performance improvements. The AutoTokenizer class from the transformers library initializes the tokenizer, loading pre-trained BART tokenizer weights using the from_pretrained method. This tokenizer, trained on a large text corpus using Byte Pair Encoding (BPE), tokenizes new input text before passing it to the model. The AutoModelForSeq2SeqLM class initializes the pre-trained BART model, designed for sequence-to-sequence tasks like summarization and translation. Pre-trained weights are loaded from the "facebook/bart-base" checkpoint. The BART model encodes input text using an autoencoder architecture and decodes it to generate summaries. We analyzed 50 Urdu news articles, generating summaries with the BART model. To visualize performance, we plotted a graph (Fig. 6) with two lines: a blue line representing original summaries and a brown line for BART-generated summaries. Comparing these lines helps assess the model's accuracy.

RESULTS AND DISCUSSIONS

We summarize the results from three summarizers, comparing their outputs using ROUGE metrics. The AutoTokenizer class from the transformers library initializes the tokenizer, with pre-trained BART tokenizer weights loaded using the from pretrained method. This tokenizer, trained on a large corpus using Byte Pair Encoding (BPE), tokenizes input text before model processing. The AutoModelForSeq2SeqLM class initializes the pre-trained BART model, designed for sequence-to-sequence tasks like summarization and translation. Pre-trained weights are loaded from the "facebook/bart-base" checkpoint. Using an autoencoder architecture, the BART model encodes input text and decodes it to generate summaries. We analyzed 50 Urdu news articles, generating summaries with the BART model. Results were visualized in Fig. 4, with a blue line representing original summaries and a brown line for BART-generated summaries. Comparing these lines assesses model accuracy. Additionally, the AutoTokenizer class initializes a tokenizer for the mT5 model, loading pre-trained weights from the csebuetnlp/mT5_multilingual_XLSum checkpoint. SentencePiece, a sub-word tokenization algorithm, tokenizes new input text using this pre-trained tokenizer, which is trained on a large multilingual corpus.

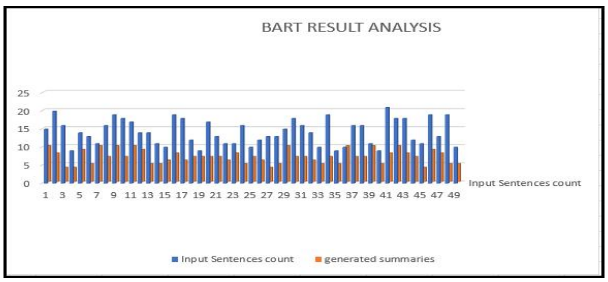


Fig. 4. BART ANALYSIS ON ABSTRACTIVE SUMMARIZATION.

After creating the tokenizer object, new input text can be tokenized before being passed to the model. The pre-trained mT5 model, trained on a large corpus using an autoencoder architecture, encodes input text and decodes its representation to generate summaries. This enables it to produce summaries for any new input text. We evaluated the T5 model's performance by summarizing 50 Urdu news articles. The results were visualized in a graph in Fig. 5, where the blue line represents the original summaries and the brown line represents the T5-generated summaries. By comparing these lines, the graph provides insights into the T5 model's summarization accuracy and performance

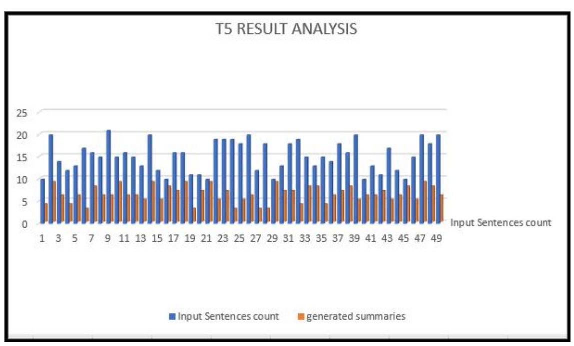


Fig. 5. T5 RESULT ANALYSIS ON ABSTRACTIVE SUMMARIZATION

We analyzed 50 Urdu news articles and used the GPT-2 model to generate summaries for each. To visualize the results, The graph illustrated in Fig. 6 with two lines: a blue line representing the original summaries and a brown line for the GPT-2-generated summaries. This graph illustrates the model's summarization performance for Urdu news articles. By comparing the blue and brown lines, we can easily assess the accuracy and quality of the generated summaries We evaluate the improved Efficient GPT-2 Summarizer model for abstractive Urdu text summarization using established metrics like coherence, informativeness, and fluency. The model undergoes rigorous testing on a dataset to assess its ability to generate reliable and informative summaries. Progress achieved by eliminating ineffective attention heads is measured against benchmarks.

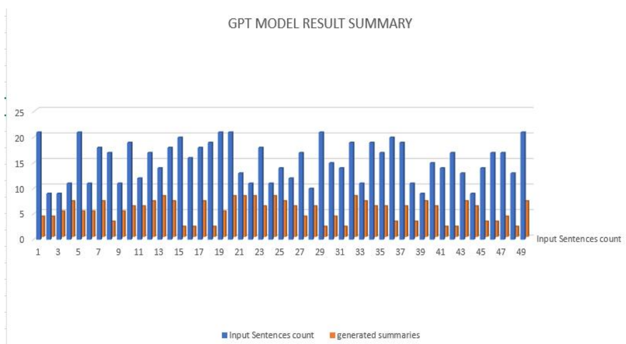


Fig. 6. GPT-2 ANALYSIS ON ABSTRACTIVE SUMMARIZATION

To validate our approach, we compare the model's performance with conventional abstractive summarization methods across diverse datasets and scenarios. This evaluation demonstrates the model's efficacy, flexibility, and superiority. Additionally, we analyzed 50 Urdu news articles, generating summaries using the BART model. Results were visualized in Fig. 7, with a blue line representing original summaries and a brown line for BART-generated summaries. Comparing these lines provides insights into the model's accuracy and summarization performance. These findings highlight the potential of our Transformer-based approach for abstractive Urdu text summarization

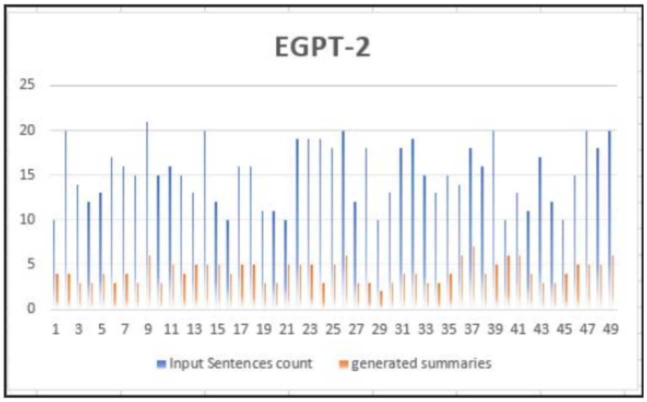


Fig. 7. EFFICIENT GPT-2 SUMMARIZER ON ABSTRACTIVE SUMMARIZATION

COMPARATIVE ANALYSIS

In this section, we are going to evaluate the summaries resulting from our text summarizers using the metric ROUGE and its variants. We are interested only in the stat F-measure for three metrics which are ROUGE-1, ROUGE-2, and ROUGE-L. The Fig. 9(a) illustrates the original paragraph to be summarized, Fig. 9(b) illustrate the summaries generated by BART model, Fig. 9(c) illustrates the summaries generated by T5, Fig. 9(d) illustrates the summaries generated by GPT and finally Fig. 9(e) illustrates the summaries generated by our proposed model EGPT. This ensure the efficiency and preserveness of actual meaning of the original text is evaluated by each proposed model.

The Table. 2 presents the ROUGE scores for three different summarization models: GPT-2, T5, BART, and EGPT-2. Comparing the models based on their ROUGE scores, we can observe that EGPT-2 achieves the highest scores across all metrics (ROUGE-1, ROUGE-2, and ROUGE-L) for all three summaries. This indicates that EGPT-2 performs exceptionally well in capturing the essence of the original text and generating high-quality summaries. Among the other models, GPT-2 also demonstrates relatively high ROUGE scores, especially for Summary 1. T5 and BART, on the other hand, exhibit lower ROUGE scores compared to EGPT-2 and GPT-2 as illustrated in Fig. 8. It can be inferred that EGPT-2 yields the best results among the evaluated models in terms of summarization quality.

The research aimed to enhance automatic text summarization for Urdu, addressing the growing need for efficient summarization methods. While extractive summarization is well-studied, this study focused on abstractive summarization, leveraging the Urdu Fake News dataset and pre-trained models. Among the tested models—BART, T5, GPT-2, and EGPT-2—EGPT-2 achieved the highest ROUGE score (0.9694). By optimizing model components, the study improved summary accuracy and coherence. The findings highlight the significance of abstractive summarization for Urdu and similar languages. Future work includes expanding datasets and exploring advanced models like mBERT to further enhance summarization quality.

TABLE II. ROUGE SCORES FOR DIFFERENT MODELS

MODELS	ROUGE-1	ROUGE-2	ROUGE-L
	0.9494	0.9518	0.9494
GPT-2	0.8116	0.7564	0.8116
	0.8749	0.787	0.8749
	0.2172	0.0982	0.1647
T5	0.4999	0.3583	0.4999
	0.4873	0.2999	0.3865
	0.3146	0.1929	0.3146
BART	0.3711	0.2033	0.3711
	0.2075	0.0133	0.1886
	0.9894	0.9718	0.9694
EGPT	0.9216	0.9064	0.9016
	0.9549	0.957	0.9149

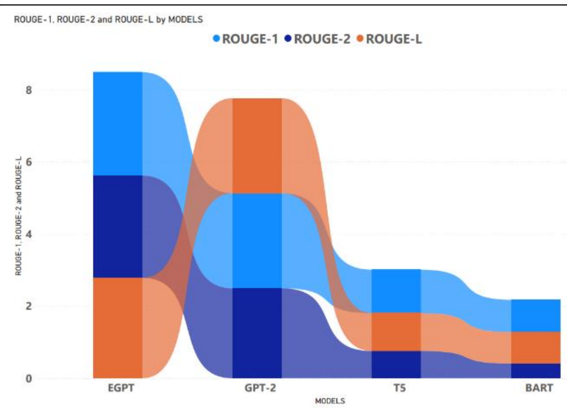
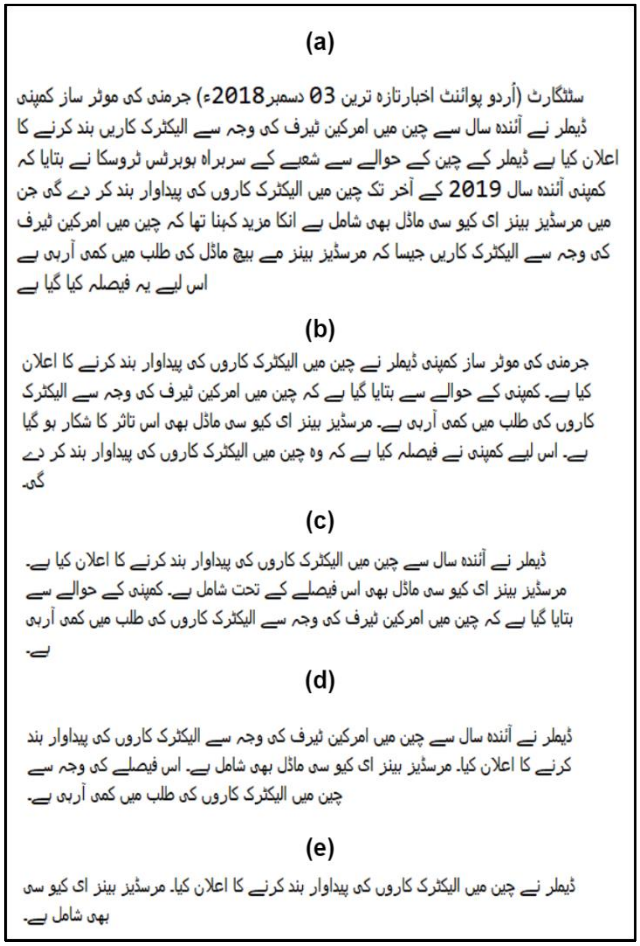


Fig. 8. ROUGE SCORES OF PROPOSED MODELS


(a)

سٹٹگارٹ (اُردو پوائنٹ اخبارتازہ ترین 03 دسمبر2018ء) جرمنی کی موٹر ساز کمپنی
ڈیملر نے آئندہ سال سے چین میں امرکین ٹیرف کی وجہ سے الیکٹرک کاریں بند کرنے کا
اعلان کیا ہے ڈیملر کے چین کے حوالے سے شعبے کے سربراہ ہوبرٹس ٹروسکا نے بتایا کہ
کمپنی آئندہ سال 2019 کے آخر تک چین میں الیکٹرک کاروں کی پیداوار بند کر دے گی جن
میں مرسڈیز بینز ای کیو سی ماڈل بھی شامل ہے انکا مزید کہنا تھا کہ چین میں امرکین ٹیرف
کی وجہ سے الیکٹرک کاریں جیسا کہ مرسڈیز بینز مے بیچ ماڈل کی طلب میں کمی آرہی ہے
اس لیے یہ فیصلہ کیا گیا ہے

(b)

جرمنی کی موٹر ساز کمپنی ڈیملر نے چین میں الیکٹرک کاروں کی پیداوار بند کرنے کا اعلان
کیا ہے۔ کمپنی کے حوالے سے بتایا گیا ہے کہ چین میں امرکین ٹیرف کی وجہ سے الیکٹرک
کاروں کی طلب میں کمی آرہی ہے۔ مرسڈیز بینز ای کیو سی ماڈل بھی اس تاثر کا شکار ہو گیا
ہے۔ اس لیے کمپنی نے فیصلہ کیا ہے کہ وہ چین میں الیکٹرک کاروں کی پیداوار بند کر دے
گی۔

(c)

ڈیملر نے آئندہ سال سے چین میں الیکٹرک کاروں کی پیداوار بند کرنے کا اعلان کیا ہے۔
مرسڈیز بینز ای کیو سی ماڈل بھی اس فیصلے کے تحت شامل ہے۔ کمپنی کے حوالے سے
بتایا گیا ہے کہ چین میں امرکین ٹیرف کی وجہ سے الیکٹرک کاروں کی طلب میں کمی آرہی
ہے۔

(d)

ڈیملر نے آئندہ سال سے چین میں امرکین ٹیرف کی وجہ سے الیکٹرک کاروں کی پیداوار بند
کرنے کا اعلان کیا۔ مرسڈیز بینز ای کیو سی ماڈل بھی شامل ہے۔ اس فیصلے کی وجہ سے
چین میں الیکٹرک کاروں کی طلب میں کمی آرہی ہے۔

(e)

ڈیملر نے چین میں الیکٹرک کاروں کی پیداوار بند کرنے کا اعلان کیا۔ مرسڈیز بینز ای کیو سی
بھی شامل ہے۔

Fig. 9. ILLUSTRATES THE DIFFERENCE SUMMARIES GENERATED BY MODELS

REFERENCES

[1] A. Muhammad, N. Jazeb, A. M. Martinez-Enriquez, and A. Sikander, "EUTS: Extractive Urdu Text Summarizer," in Proc. 17th Mexican Int. Conf. on Artificial Intelligence (MICAI 2018), Oct. 2018, pp. 39-44, doi: 10.1109/MICAI46078.2018.00014.

[2] S. Vijay, V. Rai, S. Gupta, A. Vijayvargia, and D. M. Sharma, "Extractive text summarisation in Hindi," in Proc. 2017 Int. Conf. on Asian Language Processing (IALP 2017), Jan. 2018, vol. 2018, pp. 318-321, doi: 10.1109/IALP.2017.8300607.

[3] S. R. Rahimi, A. T. Mozhdehi, and M. Abdolahi, "An overview on extractive text summarization," in Proc. 4th Int. Conf. on Knowledge-Based Engineering and Innovation (KBEI 2017), Jan. 2018, vol. 2018, pp. 54-62, doi: 10.1109/KBEI.2017.8324874.

[4] M. Asif, S. A. Raza, J. Iqbal, N. Perwaiz, T. Faiz, and S. Khan, "Bidirectional Encoder Approach for Abstractive Text Summarization of Urdu Language," in Proc. 2022 Int. Conf.

on Business Analytics for Technology and Security (ICBATS 2022), 2022, doi: 10.1109/ICBATS54253.2022.9759026.

[5] A. Nawaz, M. Bakhtyar, J. Baber, I. Ullah, and W. Noor, "Extractive text summarization models for the Urdu language," Inf. Process. Manag., vol. 57, no. 6, pp. 102383, 2020.

[6] J. Jiang et al., "Enhancements of Attention-Based Bidirectional LSTM for Hybrid Automatic Text Summarization," IEEE Access, vol. 9, pp. 123660-123671, 2021, doi: 10.1109/ACCESS.2021.3110143.

[7] E. Egonmwan and Y. Chali, "Transformer-based Model for Single Documents Neural Summarization," in Proc. 3rd Workshop on Neural Generation and Translation, EMNLP-IJCNLP 2019, 2019, pp. 70-79, doi: 10.18653/V1/D19-5607.

[8] M. Lewis et al., "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension," Oct. 2019, doi: 10.48550/arxiv.1910.13461.

[9] S. Chopra, M. Auli, and A. M. Rush, "Abstractive sentence summarization with attentive recurrent neural networks," in Proc. Conf. North American Chapter Assoc. for Computational Linguistics: Human Lang. Technol., pp. 93-98, 2016.

[10] K. Filippova, E. Alfonseca, C. A. Colmenares, Ł. Kaiser, and O. Vinyals, "Sentence compression by deletion with LSTMs," in Proc. Conf. Empirical Methods in Natural Lang. Process., pp. 360-368, 2015.

[11] J. Khyat, S. S. Lakshmi, and M. U. Rani, "Hybrid Approach for Multi-Document Text Summarization by N-gram and Deep Learning Models," [Online]. Available: [insert link if available].

[12] A. Burney, B. Sami, N. Mahmood, and Z. Azam, "Urdu text summarizer using sentence weight algorithm for word processors," Academia.edu, 2012. [Online]. Available: [insert link if available].

[13] M. Barbella and G. T., "Rouge Metric Evaluation for Text Summarization Techniques," SSRN, May 2023. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4120317.

[14] I. Khan et al., "Urdu Language Text Summarization using Machine Learning," J. Computing & Biomedical Informatics, vol. 8, no. 9, pp. 1-10, 2024.

[15] L. Kaleem, A. U. Rahman, and M. Moetesum, "RUATS: Abstractive Text Summarization for Roman Urdu," in Proc. Document Analysis Systems, 2024, pp. 16, Springer.

[16] A. Faheem, F. Ullah, M. S. Ayub, and A. Karim, "UrduMASD: A Multimodal Abstractive Summarization Dataset for Urdu," in Proc. 2024 Joint Int. Conf. on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), Torino, Italy, pp. 17245-17253, 2024. [Online]. Available: https://aclanthology.org/2024.lrec-main.1498.

[17] H. Raza and W. Shahzad, "End to End Urdu Abstractive Text Summarization With Dataset and Improvement in Evaluation Metric," IEEE Access, vol. 12, pp. 40311-40324, 2024, doi: 10.1109/ACCESS.2024.3377463.

[18] S. A. Chowdhury et al., "Improving Arabic Text Categorization Using Transformer Training Diversification," pp. 226-236, 2020.

[19] M. Barbella and T.-A. G., "Rouge Metric Evaluation for Text Summarization Techniques," SSRN, 2023. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4120317.

[20] A. M. Rush et al., "A neural attention model for abstractive sentence summarization," arXiv preprint, arXiv:1509.00685, 2015.

[21] S. Chopra, M. Auli, and A. M. Rush, "Abstractive sentence summarization with attentive recurrent neural networks," in Proc. Conf. North American Chapter Assoc. for Computational Linguistics: Human Lang. Technol., pp. 93-98, 2016.

[22] J. Gu, Z. Lu, H. Li, and V. O. Li, "Incorporating copying mechanism in sequence-to-sequence learning," arXiv preprint, arXiv:1603.06393, 2016.

[23] A. See, P. J. Liu, and C. D. Manning, "Get to the point: Summarization with pointer-generator networks," arXiv preprint, arXiv:1704.04368, 2017.

[24] S. Gehrmann, Y. Deng, and A. M. Rush, "Bottom-up abstractive summarization," arXiv preprint, arXiv:1808.10792, 2018.

[25] J. Lin, X. Sun, S. Ma, and Q. Su, "Global encoding for abstractive summarization," arXiv preprint, arXiv:1805.03989, 2018.

[26] J. Niu, M. Sun, J. J. Rodrigues, and X. Liu, "A novel attention mechanism considering decoder input for abstractive text summarization," in Proc. IEEE Int. Conf. on Communications (ICC), pp. 1-7, 2019.

[27] F. Nan, R. Nallapati, B. Zhou, and E. Goot, "Entity-level factual consistency of abstractive text summarization," arXiv preprint, arXiv:2102.09130, 2021.

[28] W. Kryściński, R. Paulus, C. Xiong, and R. Socher, "Improving abstraction in text summarization," arXiv preprint, arXiv:1808.07913, 2018.

[29] Y. Zhao, Z. Luo, and A. Aizawa, "A language model based evaluator for sentence compression," in Proc. 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 170-175, 2018.

[30] K. Filippova, E. Alfonseca, C. A. Colmenares, Ł. Kaiser, and O. Vinyals, "Sentence compression by deletion with LSTMs," in Proc. Conf. Empirical Methods in Natural Language Processing, pp. 360-368, 2015.

[31] S. A. Chowdhury et al., "Improving Arabic Text Categorization Using Transformer Training Diversification," pp. 226-236, 2020.

[32] M. Barbella and T.-A. G., "Rouge Metric Evaluation for Text Summarization Techniques," SSRN, 2023. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4120317.

[33] M. Ibrar et al., "Econnoitering Data Protection and Recovery Strategies in the Cyber Environment: A Thematic Analysis," International Journal for Electronic Crime Investigation, vol. 8, Dec. 2024, doi: 10.54692/ijeci.2024.0804216.

[34] S. Chopra, M. Auli, and A. M. Rush, "Abstractive sentence summarization with attentive recurrent neural networks," in Proc. Conf. North American Chapter Assoc. for Computational Linguistics: Human Lang. Technol., pp. 93-98, 2016.

[35] J. Gu, Z. Lu, H. Li, and V. O. Li, "Incorporating copying mechanism in sequence-to-sequence learning," arXiv preprint, arXiv:1603.06393, 2016.

[36] A. See, P. J. Liu, and C. D. Manning, "Get to the point: Summarization with pointer-generator networks," arXiv preprint, arXiv:1704.04368, 2017.

[37] S. Riaz et al., "Software Development Empowered and Secured by Integrating A DevSecOps Design," Journal of Computing & Biomedical Informatics, p. 02, Mar. 2025, doi: 10.56979/802/2025..

[38] M. W. Iqbal et al., "Meta-analysis and investigation of usability attributes for evaluating operating systems," Migration Letters, vol. 21, pp. 1363-1380, 2024.

[39] J. Niu, M. Sun, J. J. Rodrigues, and X. Liu, "A novel attention mechanism considering decoder input for abstractive text summarization," in Proc. IEEE Int. Conf. on Communications (ICC), pp. 1-7, 2019.

[40] F. Nan, R. Nallapati, B. Zhou, and E. Goot, "Entity-level factual consistency of abstractive text summarization," arXiv preprint, arXiv:2102.09130, 2021.

[41] W. Kryściński, R. Paulus, C. Xiong, and R. Socher, "Improving abstraction in text summarization," arXiv preprint, arXiv:1808.07913, 2018.

[42] Azhar, Z. (2024). Blockchain as a Catalyst for Green and Digital HR Transformation: Strategies for Sustainable Workforce Management. *Open Access Library Journal*, *11(9)*, 1-22

[43] Azhar, Z. (2024). The Role of Chatbots in Enhancing Job Seekers' and Employee Experience: A Case Study on CV Warehouse. *The Journal of Social Sciences Research*, *10(4)*, 23-35.

[44] Azhar, Z., Nawaz, H., Malik, A. S., & Zaidi, M. H. (2022). Strategic Impact of Cloud Computing on HR Transformation. *International Journal of Social Science & Entrepreneurship*, *2(2)*, 546–576. https://doi.org/10.58661/ijsse.v4i4.336

[45] Azhar, Z., & Imran, M. (2024). Ethical Considerations in the Adoption of Artificial Intelligence in Human Resource Management: A Comprehensive Review. *Journal of Emerging Technologies and Innovative Research (JETIR)*, *11*(8).