IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 19 February 2024, accepted 11 March 2024, date of publication 19 March 2024, date of current version 6 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3378300

RESEARCH ARTICLE

Abstractive Text Summarization for the Urdu Language: Data and Methods

MUHAMMAD AWAIS AND RAO MUHAMMAD ADEEL NAWAB
Department of Computer Science, COMSATS University Islamabad, Lahore Campus, Lahore 54000, Pakistan
Corresponding author: Muhammad Awais (sp19-pcs-012@cuilahore.edu.pk)

ABSTRACT The task of abstractive text summarization aims to automatically generate a short and concise summary of a given source article. In recent years, automatic abstractive text summarization has attracted the attention of researchers because large volumes of digital text are readily available in multiple languages on a wide range of topics. Automatically generating precise summaries from large text has potential application in the generation of news headlines, a summary of research articles, the moral of the stories, media marketing, search engine optimization, financial research, social media marketing, question-answering systems, and chatbots. In literature, the problem of abstractive text summarization has been mainly investigated for English and some other languages. However, it has not been thoroughly explored for the Urdu language despite having a huge amount of data available in digital format. To fulfill this gap, this paper presents a large benchmark corpus of 2,067,784 Urdu news articles for the Urdu abstractive text summarization task. As a secondary contribution, we applied a range of deep learning (LSTM, Bi-LSTM, LSTM with attention, GRU, Bi-GRU, and GRU with attention), and large language models (BART and GPT-3.5) on our proposed corpus. Our extensive evaluation on 20,000 test instances showed that GRU with attention model outperforms the other models with ROUGE-1 = 46.7, ROUGE-2 = 24.1, and ROUGE-L = 48.7. To foster research in Urdu, our proposed corpus is publically and freely available for research purposes under the Creative Common Licence.

INDEX TERMS Abstractive text summarization, BART, corpus, deep learning models, GPT-3.5, large language models, Urdu.

I. INTRODUCTION

Abstractive Text Summarization (ATS) is a task that aims to generate an automatic summary that includes words and phrases representing the most important information from the source text [1]. Due to recent technological developments, large amounts of digital data are generated every day. Extracting and combining useful information from large volumes of data is a non-trivial task. Automatic generation of abstractive summaries can help us to extract and combine useful information from one or more sources. ATS has an assortment of real-world applications, including the generation of news headlines, a summary of research articles, the moral of the stories, media marketing, search engine optimization, financial research, social media marketing, question-answering systems, and chatbots.

The problem of ATS has been mainly explored for English and some other languages [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14]. On the other hand, the application of ATS on South Asian languages, notably Urdu, has not been studied well. Only one study on the Urdu ATS has been published, and it provides a corpus of only 50 Urdu news articles [15]. As can be noted the corpus size is very small, and thus cannot be used to develop efficient Urdu ATS systems. This study intends to fill this gap by creating a large benchmark corpus for the Urdu ATS task.

Urdu is one of the most frequently spoken languages in South Asia, and it has more than 170 million speakers around the world.[1] Persian, Arabic, and other South Asian

The associate editor coordinating the review of this manuscript and approving it for publication was Nikhil Padhi.

[1] According to Ethnologue: www.ethnologue.com/language/urd Last Visited: 25-Sep-2022.

languages [16] have all influenced Urdu language's lexicon and syntax. Urdu is an Indo-Aryan language that has a rich and diverse morphology, and some of its words (nouns and verbs) may have up to 40 variants, making it difficult to mechanically analyze [17]. Even though Urdu is a widely spoken language and a considerable amount of digitized Urdu text is available, it still remains a low-resource language for Natural Language Processing (NLP) research. However, over the last few years, the research community has focused its attention on developing standard computational resources and methods to foster research in the Urdu Lagrange [18]. The presence of various benchmark corpora is critical for the development of tools and strategies for a variety of NLP activities [19].

We need standard assessment resources to develop, evaluate, and compare Urdu ATS systems. The primary goals of this study are twofold: (1) to develop a large benchmark corpus for the Urdu ATS task, and (2) to develop, apply, evaluate, and compare state-of-the-art baseline deep learning models on the proposed corpus. For the first goal, this research work proposes a large benchmark corpus of approximately 2.067 million (2,067,784) Urdu news articles for the Urdu ATS task (hereafter called UATS-23 corpus). These Urdu news articles have been taken from various domains, including sports, entertainment, national, international, business, column, science, technology, crime, health, and science. In order to achieve the second goal, we have developed, implemented, and applied six state-of-the-art baseline deep learning models including LSTM, Bi-LSTM, LSTM with Attention, GRU, Bi-GRU, GRU with Attention, and two state-of-the-art large language models, i.e., Bidirectional Auto-Regressive Transformers (BART) and Generative Pre-trained Transformer (GPT3.5) on our proposed UATS-23 corpus.

Our proposed UATS-23 corpus will, in our opinion, serve to: (1) foster research in a low-resourced language i.e., Urdu, (2) directly compare the existing state-of-the-art techniques for the Urdu ATS challenge, (3) develop, evaluate, and compare new techniques for the Urdu ATS task, (4) develop Urdu word embedding models, and (5) other Urdu NLP tasks such as Urdu news articles categorization, and clustering of Urdu news articles etc.

The organization of this paper is as follows. Section II highlights existing corpora and methods for the ATS task. Section III explains the corpus creation process for the proposed UATS-23 corpus. Section IV describes state-of-the-art baseline deep learning models and large language models that were used on the proposed UATS-23 corpus. Section V presents the experimental setup. Section VI discusses results and their analysis. Finally, Section VII concludes the paper.

II. RELATED WORK

In literature, a number of studies have explored the development of standard evaluation resources and techniques for the ATS task. We provide a thorough analysis of the available ATS corpora, methodologies, and techniques below.

A. CORPORA FOR ATS

One of the remarkable efforts to foster research in the field of text summarization is a series of competitions (or shared tasks) organized under the umbrella of Document Understanding Conference (DUC[2]) / Text Analysis Conference (TAC) [2]. The most important end result of these competitions is the creation of a set of gold standard corpora for the Extractive Text Summarization (ETS) and the ATS tasks for both single and multi-document tasks. The DUC was organized from 2001 to 2007, and the text summarization corpora were developed for the English and Arabic languages. The abstractive text summarization corpora were created by asking domain experts to manually write summaries of source documents. The size of corpora developed for the DUC is small and varies from 600 to 1250 documents [3].

In addition to the DUC corpora, researchers have made an effort to develop standard evaluation resources for the ATS task. In a study, [4] developed Cable News Network and Daily Mail (CNN/DM) corpus for the English ATS task. The CNN/DM corpus is constructed by modifying an existing corpus developed for the question-answering task [5]. This corpus comprises of 3,11,672 news articles and their corresponding multi-summaries. The authors applied RNN encoder-decoder with hierarchical and temporal attention. The best results were obtained using RNN with temporal attention model with ROUGE-1 = 35.4, ROUGE-2 = 13.3, and ROUGE-L = 32.6.

In [6], authors proposed the New York Times (NYT) corpus for automatic text summarization, metadata extraction, information retrieval, and extraction tasks. The NYT corpus contains a total of 1.8 million English news articles extracted from various online sources. The NYT corpus was annotated for the ETS task, comprising of 6,54,759 English news articles and their corresponding summaries. These summaries are written by library scientists and are mainly used for extractive text summarization tasks [8]. Authors [8] applied Bi-LSTM based multiple extractive and compressive summarizer models on CNN/DM [4] and NYT [6] corpora. The best results (on NYT corpus) were obtained using joint extractive and compressive summarizer approach with ROUGE-1 = 45.5, ROUGE-2 = 25.3, and ROUGE-L = 38.2. Another research group used NYT corpus for the ATS task for the first time by using an article-abstract pair of the NYT corpus [9]. The authors applied maximum likelihood and reinforcement learning methods with and without intra-attention techniques on CNN/DM and NYT corpora. The best results of ROUGE were obtained on the NYT corpus using reinforcement learning with an intra-attention approach (ROUGE-1 = 41.1, ROUGE-2 = 15.7, and ROUGE-L = 39.0).

In [10], the authors proposed a newsroom corpus for the English ATS task. The newsroom corpus contains more than 1.3 million English articles and their corresponding

[2]https://duc.nist.gov/ Last Visited: 25-Sep-2022.

summaries. Newsroom corpus is significantly diverse because the articles are written by different authors of multiple news domains and their corresponding summaries are written for the HTML and social media metadata. The newsroom corpus is further divided into three subsets to be used for the ATS, ETS, and mixed ATS and ETS tasks. The best results were obtained using the pointer-generator model on all subsets of the newsroom corpus (ROUGE-1 = 39.1, ROUGE-2 = 27.9, and ROUGE-L = 36.1 for ETS task, ROUGE-1 = 25.4, ROUGE-2 = 11.0, and ROUGE-L = 21.0 for mixed ATS and ETS task, and ROUGE-1 = 14.6, ROUGE-2 = 02.2, and ROUGE-L = 11.4 for the ATS task).

In [11], authors proposed the WikiHow corpus for the English ATS task. The WikiHow corpus comprises 2,04,004 articles written by ordinary people to capture the writing style in a broader aspect. The corpus is constructed by extracting the articles from the WikiHow data dump.[3] The best results were obtained using a pointer generator with a coverage model on WikiHow corpus (ROUGE-1 = 28.5, ROUGE-2 = 09.2, and ROUGE-L = 26.5).

In [12], authors annotated the Gigaword[4] corpus for the ATS task. The Gigaword corpus originally consisted of 10 million news articles without corresponding headlines (summaries). Therefore, the authors in [12] used the subset of data and annotated it by extracting the first line of the article and treating it as a summary for the remaining article. The final version of the Gigaword corpus for ATS consists of 3.8 million news articles and summary pairs for the English language.

Researchers have also explored languages other than English for the ATS task, for example, authors in [13] proposed a Persian news corpus for the ATS task. This corpus consists of 93,207 articles and their corresponding summary pairs. The authors applied various transfer learning models to the proposed Persian news corpus. The best results were obtained using transfer learning using the BERT2BERT technique on the Persian news corpus with ROUGE-1 = 44.0, ROUGE-2 = 25.0, and ROUGE-L = 37.7.

For the Chinese language ATS challenge, authors in [14] suggested a Large-scale Chinese Short Text Summarization (LCSTS) corpus. More than 2 million Chinese articles and related summaries make up the LCSTS corpus. The Chinese microblogging website's articles are crawled, together with the summaries of each, to create the corpus. The RNN with context model using a character-based input approach on the LCSTS corpus produced the greatest results, with ROUGE-1 = 29.9, ROUGE-2 = 17.4, and ROUGE-L = 27.2.

In [15], authors have proposed an Urdu summary corpus for the Urdu ATS task. The corpus comprises 50 articles and their corresponding human-written abstractive summaries. The corpus is constructed by crawling articles from several online sources, such as news portals and blogs. The summaries are then manually written by domain experts. In [19], the authors extend this Urdu summary corpus for the ETS task. The corpus is constructed by selecting the most relevant sentences from the source document by the domain experts. The corpus comprises of 600 news articles and their corresponding summaries. The best results were obtained using the weighted term technique with ROUGE-1 = 37.0, and ROUGE-2 = 65.0.

[3]https://www.wikihow.com/Main-Page Last Visited: 25-Sep-2022
[4]https://catalog.ldc.upenn.edu/LDC2003T05 Last Visited: 25-Sep-2022

B. SEQUENCE-TO-SEQUENCE METHODS FOR ATS

In literature, researchers have also developed techniques and methods by using the above-mentioned corpora for the ATS task. In [20], the authors proposed a convolution encoder model with an attention mechanism to deal with sentence-level summarization tasks. The authors applied the proposed technique on DUC-2004 [2] and Gigaword [12] corpora. The best results were obtained using the attention-based convolution encoder model on Gigaword corpus with ROUGE-1 = 31.0, ROUGE-2 = 12.6, and ROUGE-L = 28.3.

In [21], authors applied various convolution RNN and LSTM encoder-decoder models on Gigaword [12] corpus and tested on both Gigaword and DUC-2004 [2] corpora. The recurrent attention model outperforms the LSTM attention models on the Gigaword corpus for the ATS task. The authors report results on both the DUC-2004 and the Gigaword test corpora, and the best performance was reported on Gigaword corpus with ROUGE-1 = 33.7, ROUGE-2 = 15.9, and ROUGE-L = 31.1.

In [22], the authors applied a copy mechanism using sequence-to-sequence RNN to deal with the out-of-vocabulary issue on LCSTS [14]. The out-of-vocabulary issue was addressed by replicating a certain segment of the input sequence in the output sequence. The authors applied the proposed technique at the word and character level on the LCSTS corpus. The best results were obtained on the word level with ROUGE-1 = 35.0, ROUGE-2 = 22.3, and ROUGE-L = 32.0.

In [23], the authors incorporated the pointer-generator coverage model which preserves the previous attention history with the sequence model to avoid repetition in the generated summaries. The authors applied attention-based baseline and pointer-generator models on CNN/Daily Mail [4] corpus and the best results were obtained using pointer-generator with coverage model with ROUGE-1 = 39.5, ROUGE-2 = 17.2, and ROUGE-L = 36.3.

In [24], the authors introduced the content selection method along with attention sequence models. The aim is to select the most appropriate words for the abstractive summary. The authors applied various baseline RNN, and pointer-generator models with attention along with the proposed bottom-up summarization approach on the CNN/Daily Mail [4] and NYT [6] corpora. The best results were obtained by using a bottom-up summarization approach with ROUGE-1 = 41.2, ROUGE-2 = 18.6, and ROUGE-L = 38.3.

In [25], the authors applied a global encoding model to eliminate recurrence in the generated abstractive summary.

To keep the more semantically relevant information, the authors used the global encoding at the encoder side of the encoder-decoder deep learning model. The authors applied the proposed technique along with other baseline techniques on the LCSTS [14] and Gigaword [12] corpora. The best results were obtained on the LCSTS corpus by using the proposed global encoding model with ROUGE-1 = 39.4, ROUGE-2 = 26.9 and ROUGE-L = 36.5.

In [26], 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 [27], the authors strive to deal with ensuring the factual consistency of the generated abstractive summary. The authors proposed a new method by filtering the training data to ensure the factual consistency of the abstractive summary. The authors applied the transfer learning models, i.e., BERT, and Pegasus on the newsroom [10], Xsum, and the CNN/Daily Mail [4] corpora. The best ROUGE results were obtained using the Transfer learning approach with a pre-trained BERT model on the Xsum corpus with ROUGE-1 = 45.6, ROUGE-2 = 22.5, and ROUGE-L = 37.2.

In [28], the authors decompose the decoder to extract the most pertinent sentences from the source document and then use language models to the resultant abstractive summaries to further improve the quality. The authors applied various baseline techniques along with proposed reinforcement learning with language modeling techniques on CNN/Daily Mail [4] corpus. The best results were obtained using reinforcement learning with language modeling techniques on the CNN/Daily Mail corpus with ROUGE-1 = 40.1, ROUGE-2 = 17.3, and ROUGE-L = 37.5.

In [29] authors applied the sequence-to-sequence models with an attention mechanism for the sentence abstraction for the ATS task. The authors applied the proposed technique on the Gigaword [12] and Google [30] corpora. The authors changed the conventional evaluation methodology from ROUGE and the best results were obtained on the proposed technique on Google corpus with F_1 = 85.1, RASP-F_1 = 82.3, and Compression Ratio = 0.4.

In [31], the authors proposed a novel abstractive summary form that makes use of a treebank's most recent development in abstract meaning representation. In this structure, the article text is parsed to a bunch of abstract meaning representation charts, the diagrams are changed into a rundown chart, and the text is produced from the synopsis diagram afterward. The authors applied the proposed technique on the DUC and TAC [2] corpora with custom annotations. The best results were obtained using the JAMR approach on the DUC and TAC corpora with ROUGE-1 *precision* = 51.2, *recall* = 40.0, and F_1 = 44.7.

In [32] authors present the Span-Fact suite genuine rectification model that uses information gained from question noting models to make revisions in framework produced abstractive summaries through range determination. The authors applied the transfer learning models on the CNN/Daily Mail [4], XSum, and Gigaword [12] corpora and the best ROUGE results were obtained using Transfer learning along with span-fact suite genuine rectification model on CNN/Daily Mail corpus with ROUGE-1 = 41.8, ROUGE-2 = 19.4, and ROUGE-L = 38.9.

In [33], the authors proposed an ATS model based on the sequence-to-sequence RNN model for the Arabic [34] language. In the proposed model, the encoder consists of bi-directional LSTM and the decoder consists of uni-directional LSTM with global attention. The authors applied the proposed RNN-based sequence-to-sequence technique on Arabic summaries collected from several sources. Furthermore, the authors proposed a new evaluation mechanism along with the ROUGE-1, which are ROUGE1-NOORDER, ROUGE1-STEM, and ROUGE1-CONTEXT. The best results were obtained with the proposed LSTM-based seq-to-seq model with ROUGE-1 = 38.4, ROUGE1-NOORDER = 46.2, ROUGE1-STEM = 52.6 and ROUGE1-CONTEXT = 58.1.

C. TRANSFORMERS FOR ATS

In [35], the authors proposed a transfer learning-based architecture to improve the quality of the abstractive summary for long-length articles. This upgrade is improved by consolidating the entity-level information and infusing primary word information from the Wikidata information diagram into the transformer-XL with an entity transfer learning model. The authors applied the transfer learning models on the CNN/Daily Mail [4] corpus and the best ROUGE results were obtained using transformer-XL-entity-wilkidata word embedding model with ROUGE-1 = 33.8, ROUGE-2 = 12.5, and ROUGE-L = 31.2.

In [36] authors fine-tuned the GPT-3.5 on the Russian news corpus (Gazeta corpus) for the ATS task. The authors applied the proposed technique on the Gazeta corpus. The authors fine-tuned the ruGPT3Small model. The authors changed the conventional evaluation methodology from ROUGE and the best results were obtained on the proposed technique on Gazeta corpus with BERTscore: precision = 0.87, BERTscore: recall = 0.90, and BERTscore: F_1 = 0.89.

In [37] authors evaluate the GPT-3.5 against various fine-tuned models for the ATS task. The authors fine-tuned the *text* − *DaVinci* − 002 model of GPT-3.5 on CNN/Daily Mail [4], and XSum corpora. The authors change the conventional evaluation methodology from ROUGE and the best results were obtained on the proposed technique on XSum corpus with BERTscore = 0.90.

In [38] authors proposed the Russian news corpus (Gazeta corpus) and evaluate it using mBART transformer for the ATS task. The authors used the pre-trained model of mBART, which was pre-trained on the monolingual corpora for 25 languages, including Russian. The authors applied fine-tuned mBART on the Gazeta corpus and the best results were obtained on the proposed corpus with fine-tuned mBART transformer with ROUGE-1 = 32.1, ROUGE-2 = 14.2, and ROUGE-L = 27.9.

The hierarchical BART (Hie-BART), which encapsulates the hierarchical structures of documents, was proposed by the authors in [39]. The interaction between sentence-level and word-level information was added by the authors to the BART model. The CNN/Daily Mail [4] corous was used by the authors to assess the suggested transformer for the ATS task, and the best performance was obtained with the score of ROUGE-1 = 44.3, ROUGE-2 = 21.3, and ROUGE-L = 41.0.

In [40] authors proposed the BART-IT transformer based on the BART architecture tailored for the Italian language. The proposed transformer is trained on large Italian language corpora, i.e., Clean Italian mC4 Corpus. It is further fine-tuned on three different Italian language corpora, i.e., FanPage, IlPost, and WITS. The authors evaluated the proposed transformer for the ATS task with all three corpora and the best results were obtained on the WITS corpus with ROUGE-1 = 42.3, ROUGE-2 = 28.8, and ROUGE-L = 38.8.

To summarize, most research on the ATS task has been conducted in English and other languages, such as Chinese [14], Persian [13], Arabic [34], and Russian [38]. The only study on Urdu ATS in the literature [15] has two major limitations: (1) the corpus is very small, with only 50 article-summary pairs, which is not enough to develop efficient Urdu ATS systems; and (2) state-of-the-art deep learning models and transformer models have not been applied to the corpus. This work proposed a large benchmark corpus of over 2.067 million summaries of Urdu news articles to address these issues. On our proposed corpus, we also created, used, and compared eight basic deep-learning models and transformers. To the best of our knowledge, no study for the Urdu ATS challenge has previously reported on the construction of a large benchmark Urdu ATS corpus and the comparison of baseline deep learning models and transformers on such a corpus.

III. CORPUS GENERATION PROCESS

Developing a large benchmark corpus for the Urdu ATS task is the primary goal of this study. The process to develop our proposed UATS-23 corpus includes raw data collection, cleaning and pre-processing of raw data, corpus characteristics, and corpus standardization. Below we describe the corpus generation process in detail.

A. RAW DATA COLLECTION

We selected the journalism domain to develop our proposed UATS-23 corpus. The choice of the Journalism field was made possible by the abundance of freely and easily accessible digital Urdu text that can be used for research on a variety of subjects.

TABLE 1. Main statistics of UATS-23 corpus.

Total news articles extracted	2,103,460
Null news articles	11,000
Duplicate news articles	24,676
Total news articles in UATS-23 corpus	2,067,784

The raw data for our proposed corpus was collected from two main sources: (1) Mendeley online digital repository of Urdu news articles [41], and (2) Urdu news articles collected from various online newspapers in Pakistan. From the first source, we found a free and publicly available collection of 1.038 million (1,038,341) Urdu news articles on various domains including showbiz, entertainment, sports, politics, national and international news. Regarding the second source, we collected Urdu news articles from the top online Urdu newspapers including Express news,[5] Daily AAj news,[6] Geo news,[7] Jang news.[8] Inquilab news,[9] Daily Aaj news[10] and Urdu news.[11] The news articles were collected from various domains, including entertainment, national, international, business, column, crime, health, and science. We extracted 1.065 million (1,065,119) Urdu news articles published between 2001 to 2021. We used a Web Crawler to extract Urdu news articles from the websites of the online Urdu newspapers. Hence, a total of approximately 2.10 million (2,103,460) Urdu news articles were extracted from two data sources.

B. DATA CLEANING AND PRE-PROCESSING

The 2.10 million (2,103,460) Urdu news articles were cleaned by removing duplicates and null values. After that, each Urdu news article was pre-processed by removing special characters, HTML tags, and tabs. In the next step, Urdu news articles were tokenized to identify correct word boundaries. After data cleaning and pre-processing, a total of 2,067,784 (approximately 2.067 million) Urdu news articles were compiled.

C. CORPUS STANDARDIZATION

All of the 2,067,784 (approximately 2.067 million) Urdu news articles were used to construct our proposed UATS-23 corpus (see Table 1). Since an Urdu news article mainly comprises of two things: (1) a headline, and (2) a detailed story,

[5] expressnews.pk Last Visited: 25-Sep-2022.
[6] dailyaaj.pk Last Visited: 25-Sep-2022.
[7] geonews.pk Last Visited: 25-Sep-2022.
[8] jangnews.pk Last Visited: 25-Sep-2022.
[9] inqalabnews.com Last Visited: 25-Sep-2022.
[10] dailyaajnews.pk Last Visited: 25-Sep-2022.
[11] urdunews.pk Last Visited: 25-Sep-2022.

TABLE 2. UATS-23 corpus characteristics.

	Source	Summary
Total number of news articles	2,067,784	2,067,784
Total tokens	44,087,300	231,985
Average no. of words	205.6 words	9.39 words
Maximum tokens	3,000 words	32 words
Minimum tokens	20 words	5 words

therefore, in our proposed corpus, the detailed description of the story is treated as the source text and the headline is treated as a summary of the source text.

The proposed UATS-23 corpus[12] has been standardized in the CSV format and made publicly available for research purposes under the Creative Commons license.[13]

D. CORPUS CHARACTERISTICS

Table 2 shows the main characteristics of the proposed UATS-23 corpus. The corpus comprises approximately 2.067 million news articles. The source text and title text have an average length of 205.6 and 9.39 words respectively. The total number of tokens in the source text is 44,087,300 and the total number of words in the summary is 231,985. The maximum number of tokens in the source text and title text are 3,000 and 32 words, respectively. Whereas, the minimum number of tokens in the source text and title text are 20 and 5 words, respectively

IV. DEEP LEARNING AND TRANSFORMER TECHNIQUES FOR THE URDU ABSTRACTIVE TEXT SUMMARIZATION

To demonstrate how our proposed UATS-23 corpus can be used for the development, evaluation, and comparison of the Urdu ATS system, we have applied six baseline deep learning models including LSTM-based encoder-decoder architecture (see Section IV-A), Bi-LSTM-based encoder-decoder architecture (see Section IV-B), GRU-based encoder-decoder architecture (see Section IV-C), Bi-GRU-based encoder-decoder architecture (see Section IV-D), LSTM-based encoder-decoder architecture with attention (see Section IV-E), GRU-based encoder-decoder architecture with attention (see Section IV-E), Bidirectional Auto-Regressive Transformers (BART) (see Section IV-F), and Generative Pre-trained Transformer (GPT-3.5.5) (see Section IV-G). Below we describe these baseline deep-learning models and transformers in detail.

[12]A sample of 100k Urdu news articles from UATS-23 corpus can be downloaded from this link:
URL: https://drive.google.com/file/d/1HNzQDaRqb5hG-fMUXMJhsn0wEtq0VxZA/view?usp=sharing
The entire UATS-23 corpus will be made publicly available for research purposes after the acceptance of the paper.

[13]https://creativecommons.org/licenses/?lang=en Last Visited: 25-Sep-2022.

A. LONG SHORT-TERM MEMORY

LSTM networks belong to the family of RNNs that can maintain long-term dependencies, which are necessary to provide an effective abstractive summary for a certain source article. LSTM has been effectively used for various NLP applications including machine translation [42], [43], image captioning [44], and the ATS [45], [46].

For this study, the LSTM-based encoder-decoder architecture proposed by [47] was used, which comprises 512 input units, a single hidden layer, and a single input-output layer. Both the encoder and the decoder's hidden layers are made up of 512 hidden units. The reason for selecting this encoder-decoder architecture was that it has been proven to be effective for various NLP tasks including machine translation [42], [43], image captioning [44], Chinese ATS [14], Arabic ATS [34], Persian ATS [13], and the English ATS [45], [46].

For the Urdu ATS task, the LSTM-based encoder-decoder architecture was trained as follows. The Urdu source article is tokenized into tokens (words). After tokenization, each word is converted into a word dictionary, which includes words and their corresponding indices, which are fed as input to the word embedding layer. The embedding layer was implemented using a 300-dimensional Urdu pre-trained word embedding model [48] (containing 140M Urdu tokens). To avoid the word embedding model from updating, the embedding layer was frozen. Note that the words that were not present in the pre-trained word embedding model were initialized with random values using Gaussian space. After freezing the embedding layer, the output of this embedding layer was passed on to the LSTM single encoder input layer having 512 memory cells. The input layer takes a single word at a time on each time step. In addition to the embedding vector, the initial hidden state and cell vector were also passed to the memory cell of the input layer. After each time step, the encoder tries to encapsulate the information in these hidden states and cell vectors. The final hidden state vector encapsulates all the information of the given input (Urdu source article) and constructs a context vector, which is then passed to the single layer decoder having 512 memory cells, that decodes the context learned by the encoder. Note that in the training phase, the teacher-forcing approach is then used to train the decoder. Using this approach, the encoder's

current time step takes the input of the true output token (from the training data) instead of the predicted one from the previous time step.

B. BI-DIRECTIONAL LONG SHORT-TERM MEMORY

One of the major limitations of the LSTM-based encoder-decoder architecture is that it fails to capture the structure of the whole input sequence [49]. To overcome this limitation, Bi-LSTM-based encoder-decoder architecture was proposed [50]. In a Bi-LSTM-based encoder-decoder architecture, two hidden layers are integrated in opposite directions for the same input sequence. The first layer processes the information in a forward direction from the start of the sequence to the end. The second layer processes information in the reverse direction at the same time. The final forward and backward outputs are concatenated to create the encoded vector. The goal of the concatenation is to enable the output layer to utilize the important information from both the preceding and future context. This helps to attain more contextual information which results in improved learning and greater efficiency.

For the Urdu ATS task, the Bi-LSTM-based encoder-decoder architecture was trained in the same way it was done for the LSTM-based encoder-decoder architecture (see Section IV-A).

C. GATED RECURRENT UNIT

Another limitation of LSTM-based encoder-decoder architecture is that it is computationally expensive. The reason for this is that it uses three gates (input, output, and forget). To reduce the computational cost of LSTM-based encoder-decoder architecture, the GRU-based encoder-decoder architecture was proposed [51]. GRU reduces the computational cost by using two gates, which are: a reset and an update gate [46].

For the Urdu ATS task, the GRU-based encoder-decoder architecture was trained in the same way it was done for the LSTM-based encoder-decoder architecture (see Section IV-A).

D. BI-DIRECTIONAL GATED RECURRENT UNIT

The functionality of the Bi-GRU-based encoder-decoder architecture is the same as that of the Bi-LSTM encoder-decoder architecture ((see Section IV-B). All the RNN based encoder-decoder models follow the same architecture as shown in figure 2.

E. ENCODER-DECODER MODELS WITH ATTENTION

One of the major limitations of all the basic encoder-decoder architectures discussed in the previous sections is that they fail to capture the complete context of long input sequences [52]. The reason for this information loss is that they only handle fixed-length context vectors. To overcome this limitation, an attention mechanism was introduced [52]. The main goal of the attention mechanism is to generate variable-length context vectors. To achieve this goal, the attention layer is added just after the encoder hidden layer(s) to generate the context vectors of variable length. The attention layer calculates the weights between each input and output word. These weights will further decide which input sequence word is considered in the generation of context vectors.

For all the encoder-decoder architectures used in this study, a local attention mechanism was used, which selects a small subset of the source words to generate a context vector. The reason for selecting a local attention mechanism is that it is computationally inexpensive and focuses only on the specific and relevant information of the input sequence. All the RNN based encoder-decoder models with attention follow the same architecture as shown in figure 2.

These baseline deep-learning models were selected due to their established track record in text summarization tasks across various languages. By including a diverse set of baseline models, we aim to establish a comprehensive benchmark against which the performance of more advanced models can be evaluated.

F. BIDIRECTIONAL AUTO-REGRESSIVE TRANSFORMERS (BART)

BART is a model for natural language processing that Facebook AI created. BART is a denoising autoencoder that has been trained on a sizable text and code training dataset [53]. It can be used for many different things, such as question-answering, text summarization, and machine translation.

BART is based on the Transformer architecture, which has proven to be particularly successful for tasks involving natural language processing, and is the foundation upon which BART is built. BART differs from other large language models in that it uses a bidirectional encoder and a left-to-right decoder. This allows BART to learn to attend to both the past and the future context of a sentence, which is important for tasks such as machine translation and text summarization.

BART was chosen for its ability to leverage large-scale pretraining on diverse text corpora, making it adept at understanding and generating natural language across different domains. Given its strong performance in English summarization tasks [53], [54], we hypothesized that fine-tuning BART on Urdu news articles would yield high-quality summaries in the Urdu language. BART has been shown to achieve state-of-the-art results on a variety of natural language processing tasks. For example, BART has been shown to outperform BERT and RoBERTa on the machine translation task [53], and it has also been shown to outperform GPT-3.5 on the text summarization task [54].

G. GENERATIVE PRE-TRAINED TRANSFORMER (GPT-3.5)

The large language model GPT-3.5 was created by OpenAI. It is a 175 billion parameter decoder-only transformer model that was developed using a sizable dataset of text and

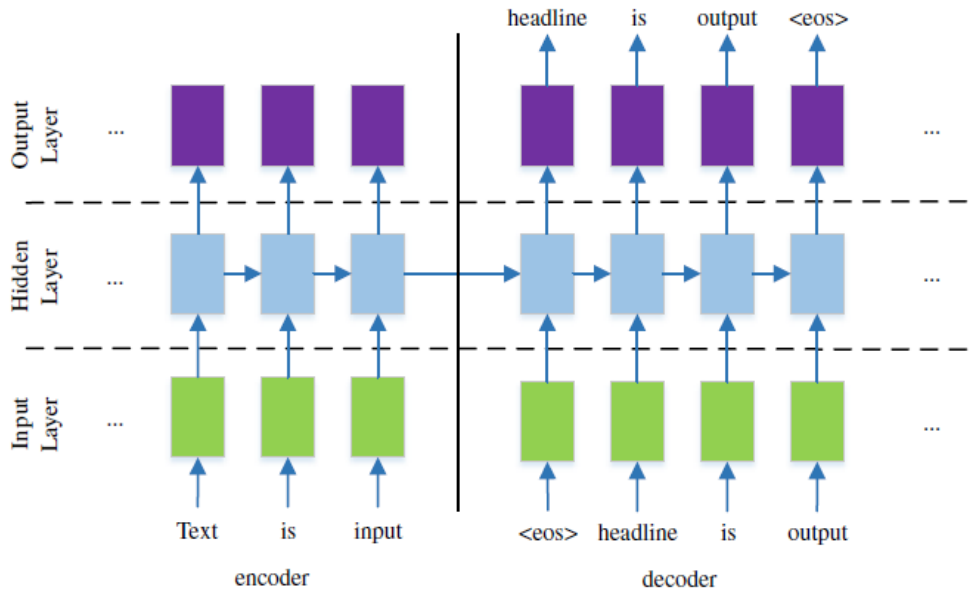

FIGURE 1. RNN based encoder-decoder architecture.

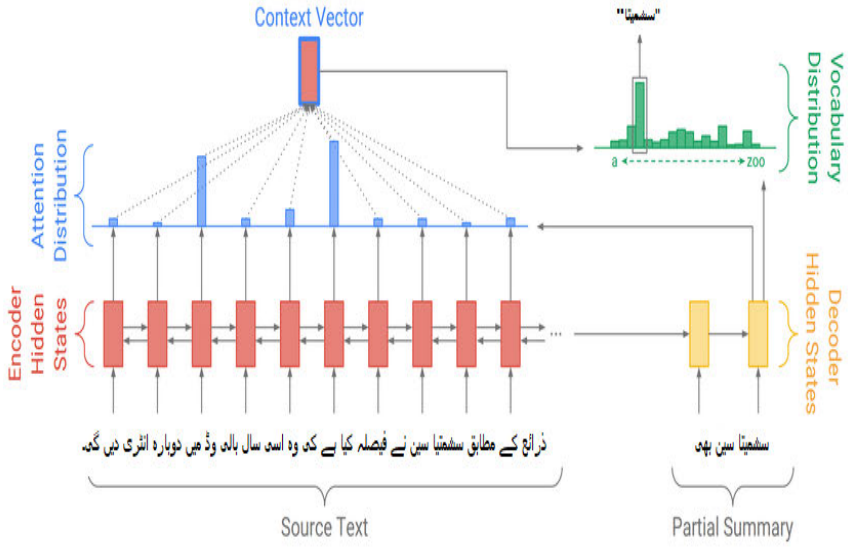

FIGURE 2. RNN based encoder-decoder architecture with attention.

code [55]. GPT-3.5 can generate text, translate languages, write various types of creative content, and provide you with well-informed answers to your queries [56].

GPT-3.5 is based on the Transformer architecture [57], which is a neural network architecture that has been shown to be very effective for natural language processing tasks. By utilizing a sparse attention mechanism [55], GPT-3.5 differentiates from other large language models and is able to recognize long-range dependencies in text. GPT-3.5 has been shown to achieve state-of-the-art results on a variety of natural language processing tasks, including text generation, translation, and question answering [55], [56]. GPT-3.5 was selected due to its impressive generative capabilities and the breadth of knowledge it has acquired through pretraining on a massive text corpus. We anticipated that fine-tuning GPT-3.5 on our Urdu news corpus would leverage its comprehensive understanding of language patterns and semantics to produce fluent and informative summaries in the Urdu language. In this work, we have used OpenAI's *text − curie − 001*, a GPT-3.5 model from the Instruct series.

V. EXPERIMENTAL SETUP

This section presents the dataset, state-of-the-art baseline deep learning models and Transformers (BART and GPT-3.5), evaluation methodology, and evaluation measures.

A. DATASET

For the experiments presented in this study, the entire UATS-23 corpus was used (containing 2,067,784 ($\simeq$ 2.067 million) Urdu news articles along with their summaries (see Section III for details)).

B. TECHNIQUES

The techniques applied on the UATS-23 corpus include LSTM-based encoder-decoder architecture (see Section IV-A), Bi-LSTM-based encoder-decoder architecture (see Section IV-B), GRU-based encoder-decoder architecture (see Section IV-C), Bi-GRU-based encoder-decoder architecture (see Section IV-D), LSTM-based encoder-decoder architecture with attention (see Section IV-E), GRU-based encoder-decoder architecture with attention (see Section IV-E), BART (see Section IV-F), and GPT-3.5 (see Section IV-G). Below we describe the parameter settings for these state-of-the-art baseline deep learning models and transformers.

For experiments, the Google VERTEX AI platform[14] having a single NVidia Tesla T4 GPU, 60 GB RAM, and 16 CORE vCPUs was used. The experiments were executed on Python 3.7 framework using the PyTorch[15] library. For the Urdu ATS problem, a word-level approach was used, where a model takes one Urdu word as input at each time step. The length of the output summary was set to 10 words because the length of more than 75% summaries in the UATS-23 corpus is 12 words or less. The input vocabulary for the encoder is 45,000 unique word types where the minimum frequency of words is set to two, and the output vocabulary size is 15,000 word types. Note that when we tried to increase vocabulary size, the out-of-memory message was generated by the NVidia Tesla T4 GPU due to its cache memory limitation.

All six deep learning models were trained using Adam optimizer with default settings of betas $(0.9, 0.999)$, $weightdecay = 0$, and $learningrate = 0.001$ and using a fixed learning rate. While LSTM and GRU do not experience vanishing gradient issues, they do continue to experience exploding gradient issues. Therefore, to avoid the exploding gradient problem, gradient clipping was used, and its value was set to 1. The dropout rate of 0.5 was applied to avoid the overfitting problem. Batches of size 64 were used for the LSTM, Bi-LSTM, and GRU, whereas batches of size 32 were used for the Bi-GRU, LSTM with attention, and GRU with attention. The cache memory limit of the GPU forced us to set the batch size to 32 for Bi-GRU, LSTM with attention, and GRU with attention. Decoder teacher forcing ratio [58] was set to 0.7%. The value for a number of epochs was set in the range 1 - 10.

We fine-tuned the BART model on 100 instances of the UATS-23 training corpus, using the same architecture as described in [53]. We set a maximum sequence length of 1024 tokens for the input documents and 120 tokens for the summaries. We used a beam search of size 2, which means that the model returns the best summary among the top 2 generated summaries. We optimized the model using the AdamW optimizer [59] with a maximum learning rate of 10^{-4} and a weight decay of 10^{-2}. The learning rate was decayed according to a decay factor after the warmup phase. This strategy allows the optimizer to take progressively smaller steps as training progresses, which helps to improve convergence and model performance.

Moreover, for our experiments, we used a pre-trained model $text-curie-001$ to fine-tune it on the Urdu ATS task. The reasons for selecting this model of GPT-3.5 are its low cost and effectiveness for the ATS task. For fine-tuning we used the default parameters of GPT-3.5. For evaluation, we set the $temperature = 0$ so the model becomes more restricted to the contents of the source text. We set $max\_tokens = 100$ to ensure that the generated summary remains concise and within a specific length limit. The top-p sampling parameter controls the diversity of the generated output. We set it to 1, which means the model only considers the most likely token at each step. This setting can make the generated output more focused and coherent. The *frequency_penalty* parameter discourages the model from repeating the same phrases or tokens too frequently. A value of 0.5 strikes a balance between encouraging variety and maintaining coherence in the generated text.

C. EVALUATION METHODOLOGY

The problem of the Urdu ATS was treated as a supervised machine learning problem. To train more accurately our state-of-the-art baseline deep learning models, the entire dataset was split using the Train/Test/Validation approach. The data used for training, testing, and validation was 2,027,784 instances, 20,000 instances, and 20,000 instances respectively. For transformers model fine-tuning we have used 100 instances for fine-tuning due to cost and computation limitations and 20,000 instances for testing to ensure the fairness in the result comparison.

The deep learning models and transformers were evaluated using averaged F_1 scores of ROUGE-1, ROUGE-2, and ROUGE-L metrics.[16] To accurately evaluate an ATS, the two aspects of the automatically generated summary are matched with the reference summary: (1) content overlap and (2) fluency of content. The ROUGE-1 and ROUGE-2 aim to compute the content overlap between reference summary and automatic summary. The results will be evaluated using the following ROUGE evaluation metrics: ROUGE-N in which we use ROUGE-1 given in eq. 1-3, and ROUGE-2 given in eq. 4-6 for unigram and bigram respectively and we also use ROUGE-L given in eq. 8-10 that depicts the longest common subsequence (LCS) as shown in eq. 7 to evaluate the fluency of the generated summary. The performance will be evaluated using Precision, Recall, and F1 score across all ROUGE metrics.

[14]https://cloud.google.com/ Last Visited: 25-Sep-2022
[15]www.pytorch.org Last Visited: 25-Sep-2022
[16]https://github.com/pltrdy/rouge Last Visited: 25-Sep-2022

Human Summary	BART Generated Summary
میکسکو میچ کے دوران امپائرکاقتل	میکسو سپورٹس ڈسک میکسکن ریاست ہڈالگومیں فٹ بال میچ کے دوران کھلاڑی نے سر سے ٹکر مار کر میچ ریفری کا قتل کردیا
خالد محمود سے اظہار تعزیت	کراچاسپورٹس رپورٹربرائٹ کلب زون چار کے پلئر خالد محمود کے بڑے بھائی ارشد محمو اور نائب صدر افضل خان تعزیت کا اظہار
محفل مشاعرہ کل ہو گی	لاہور کلچرل رپورٹر بزم عمل پاکستان لاہور کے زیراہتمام شاعر اقبال راہی کی سالگرہ کرارتان اور موقع پر ایک شام اور مشاعرہ کا انع

FIGURE 3. Examples of good BART model performance.

Human Summary	GPT-3.5 Generated Summary
سرفراز احمد انگلنڈ روانہ ہوگئے	جبکہ مارک شائرکی نمائندگی کرنے کے لئے بعد کیریبن پرمیئرلیگ میں شرکت کیلئے ویسٹ انڈزروانہ کیلئے سرفراز احمد
پاکستان نے والے فٹبالر کون	کرینٹر میں عمر صرف ۱۷ سال تھی رابرٹ پیرز کا تعلق ویلز سے اہم کرنے کا اعلان
کپاس کے داموں میں مجموعی طور پر استحکام	پاکستان میں کپاس کی باعث کاروباری حجم بھی گھنٹے پر استحکام جاری رہا

FIGURE 4. Examples of good GPT-3.5 model performance.

ROUGE-1

$$Precision(P) = \frac{\text{Number of Overlapping Unigrams}}{\text{Total words in Automatic Summary}} \quad (1)$$

$$Recall(R) = \frac{\text{Number of Overlapping Unigrams}}{\text{Total words in Reference Summary}} \quad (2)$$

$$F_1 = \frac{2(P \times R)}{P + R} \quad (3)$$

ROUGE-2

$$Precision(P) = \frac{\text{Number of Overlapping bigrams}}{\text{Total words in Automatic Summary}} \quad (4)$$

$$Recall(R) = \frac{\text{Number of Overlapping bigrams}}{\text{Total words in Reference Summary}} \quad (5)$$

$$F_1 = \frac{2(P \times R)}{P + R} \quad (6)$$

ROUGE-L

$$LCS = max\{s_1, s_2, s_3 \ldots s_n\} \quad (7)$$

$$Precision(P) = \frac{|\text{LCS}|}{\text{Total words in Automatic Summary}} \quad (8)$$

$$Recall(R) = \frac{|\text{LCS}|}{\text{Total words in Reference Summary}} \quad (9)$$

$$F_1 = \frac{2(P \times R)}{P + R} \quad (10)$$

where, s_n denotes the subsequence n.

VI. RESULTS AND ANALYSIS

Table 3 presents the results obtained by applying the six baseline deep learning models (LSTM, Bi-LSTM, LSTM with attention, GRU, Bi-GRU and GRU with attention) and two transformers (BART and GPT-3.5) on our proposed UATS-23 corpus. For each technique, the average F_1 scores are reported for ROUGE-1, ROUGE-2, and ROUGE-L.[17]

TABLE 3. Results obtained by applying various state-of-the-art baseline deep learning models on our proposed UATS-23 corpus.

Model	ROUGE		
	1	2	L
LSTM	24.0	7.2	27.0
Bi-LSTM	27.0	8.9	29.0
GRU	43.0	19.0	43.0
Bi-GRU	22.0	5.4	26.0
LSTM with Atten.	26.0	9.0	30.0
GRU with Atten.	**46.7**	**24.1**	**48.7**
BART	1.7	0.0	0.1
GPT-3.5	13.6	4.2	12.1

Overall, the best results are obtained using the GRU with attention model (F_1 score of ROUGE-1 = 46.7, ROUGE-2 = 24.1, and ROUGE-L = 48.7), which shows that this model is most suitable for the Urdu ATS task on our proposed UATS-23 corpus. The performance of the GRU with attention model is significantly higher as compared to other deep learning models. As can be noted there is not much difference between ROUGE-1 and ROUGE-L scores, which shows that the GRU with attention model generates summaries, which have both content overlap and fluency with the manual summaries. The GRU with attention model demonstrated superior performance compared to other

[17]The detailed results can be downloaded from the following link: https://drive.google.com/drive/ folders/1bUHCchr5jOfUMKz8WcvGYNFc_CSB_XYj?usp=sharing.

models, as evidenced by higher ROUGE scores and better overall summarization quality. One potential reason for this is the effectiveness of the attention mechanism in capturing important information and context from the input text. GRUs are generally better suited for capturing dependencies in sequences compared to LSTMs. In the context of Urdu text summarization, where long sentences and complex structures are common, GRUs are more effective in capturing these dependencies. GRUs might interact more efficiently with the attention mechanism, allowing the model to focus on relevant parts of the input sequence more effectively. This can lead to better summarization performance, especially when dealing with longer texts. Urdu language has specific linguistic characteristics that are better captured by GRUs compared to LSTMs. These characteristics could include syntax, morphology, or other structural elements that GRUs are more adept at learning. In addition to that, the decrease in the ROUGE of large language models is because the ROUGE only captures the content overlapping instead of contextual information. However, the manual inspection of the results of the large language models shows that it also generates meaningful summaries. The best Rouge score for English ATS for similar news corpus, i.e., NYT corpus [6] was obtained with ROUGE-1 = 45.5, ROUGE-2 = 25.3, and ROUGE-L = 38.2, whereas with our limited resources, current architecture and foundational work on Urdu we are able to achieve a ROUGE-1 = 46.7, ROUGE-2 = 24.1, and ROUGE-L = 48.7. To further increase the performance of large language models, we have to increase the fine-tuning dataset size to capture the diversity of the large-scale dataset, which is out of the scope of this work due to cost limitations. In Figures 3 and 4, we give good illustrations of large language models with summaries that are meaningful but not quite as accurate as human annotations. The optimal parameter values obtained for this model are a learning rate of 0.001, a dropout rate of 0.5, a number of epochs are 7, and a batch size of 32 with a single hidden layer. Moreover, the ROUGE results of large language models (BART and GPT-3.5) are significantly low. As is evident, our transformer model performs badly when measured using n-gram overlap measures. On the basis of this, we draw the conclusion that our transformer models produce promising outcomes in the ATS challenge. Additionally, we manually check a sample of the generated summaries and find a number of liabilities that can be fixed.

It can also be noted from these results that the overall ROUGE scores are not very high. This highlights the fact that the Urdu ATS task is a challenging task and there are still chances for performance improvement. These preliminary results are likely to serve researchers as a promising baseline to compare their techniques for the Urdu ATS task on our proposed corpus.

Future research efforts could focus on investigating the interpretability and explainability of summarization models, especially those incorporating attention mechanisms, which could provide valuable insights into how these models generate summaries and which parts of the input text they prioritize. Conduct a thorough hyperparameter search to find the optimal configuration for your GRU with an attention model. Techniques such as random search, grid search, or Bayesian optimization can help efficiently explore the hyperparameter space and identify settings that maximize performance. Further, explore incorporating linguistic features, domain-specific embeddings, or additional metadata to better capture the nuances of Urdu ATS. Moreover, an increase in fine-tuning data improves the performance of large language models, i.e., BART and GPT-3.5. The major challenge to dealing with large-scale dataset is the cost of computational resources. In the case of, large language models the cost is also a major factor in limiting experimental setup for low-resource languages. In our experience, the GPT-3.5 calculates tokens for the Urdu language differently than the English language. This highly affects the cost. In addition to that, the accessibility of GPU for a long period is also a challenging task.

VII. CONCLUSION

A large benchmark corpus for the Urdu ATS task is presented in this paper. The 2,067,784 (about 2.067 million) Urdu news items in our proposed UATS-23 corpus come from a variety of industries, including sports, entertainment, national and international news, business, columns, science, technology, crime, health, and science. To demonstrate the applicability of our proposed corpus for developing and evaluating Urdu ATS systems, we trained eight state-of-the-art baseline deep learning models and transformers on the proposed corpus, including LSTM-based encoder-decoder, Bi-LSTM-based encoder-decoder, GRU-based encoder-decoder, Bi-GRU-based encoder-decoder, LSTM-based encoder-decoder with attention, GRU-based encoder-decoder with attention, BART, and GPT-3.5. The best results were obtained using the GRU-based encoder-decoder with attention model (F_1 score of ROUGE-1 = 46.7, ROUGE-2 = 24.1, and ROUGE-L = 48.7). For the objective of promoting research in the Urdu language, our proposed corpus would be made freely and openly accessible to the researchers. Exploring multiple tokenizers and pre-trained word embedding model to capture vocabulary of the specific domain, i.e., news articles could help to improve the results further. Furthermore, for large language models, i.e., BART and GPT-3.5, an increase in fine-tuning data could increase the performance of the models.

REFERENCES

[1] M. Gambhir and V. Gupta, "Recent automatic text summarization techniques: A survey," *Artif. Intell. Rev.*, vol. 47, no. 1, pp. 1–66, Jan. 2017, doi: 10.1007/s10462-016-9475-9.

[2] P. Over, H. Dang, and D. Harman, "DUC in context," *Inf. Process. Manage.*, vol. 43, no. 6, pp. 1506–1520, Nov. 2007, doi: 10.1016/j.ipm.2007.01.019.

[3] R. D. Lins, H. Oliveira, L. Cabral, J. Batista, B. Tenorio, R. Ferreira, R. Lima, G. D. F. P. E. Silva, and S. J. Simske, "The CNN-corpus: A large textual corpus for single-document extractive summarization," in *Proc. ACM Symp. Document Eng.*, Sep. 2019, pp. 1–10, doi: 10.1145/3342558.3345388.

[4] R. Nallapati, B. Zhou, C. N. dos santos, C. Gulcehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence RNNs and beyond," 2016, *arXiv:1602.06023*.

[5] K. M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom, "Teaching machines to read and comprehend," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 1693–1701.

[6] E. Sandhaus, "The New York times annotated corpus," *Linguistic Data Consortium*, vol. 6, no. 12, 2008, Art. no. e26752. [Online]. Available: https://hdl.handle.net/11272.1/AB2/GZC6PL

[7] E. Mozzherina, "An approach to improving the classification of the New York Times annotated corpus," in *Knowledge Engineering and the Semantic Web: 4th International Conference, KESW 2013, St. Petersburg, Russia, October 7–9, 2013. Proceedings 4*. Springer, 2013, pp. 83–91, doi: 10.1007/978-3-642-41360-5_7.

[8] J. Xu and G. Durrett, "Neural extractive text summarization with syntactic compression," 2019, *arXiv:1902.00863*.

[9] R. Paulus, C. Xiong, and R. Socher, "A deep reinforced model for abstractive summarization," 2017, *arXiv:1705.04304*.

[10] M. Grusky, M. Naaman, and Y. Artzi, "Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies," 2018, *arXiv:1804.11283*.

[11] M. Koupaee and W. Y. Wang, "WikiHow: A large scale text summarization dataset," 2018, *arXiv:1810.09305*.

[12] C. Napoles, M. Gormley, and B. van Durme, "Annotated Gigaword," in *Proc. Joint Workshop Autom. Knowl. Base Construction Web-Scale Knowl. Extraction*, 2012, pp. 95–100, doi: 10.35111/mv9t-vv26.

[13] M. Farahani, M. Gharachorloo, and M. Manthouri, "Leveraging ParsBERT and pretrained MT5 for Persian abstractive text summarization," in *Proc. 26th Int. Comput. Conf., Comput. Soc. Iran (CSICC)*, Farah, Iran, Mar. 2021, pp. 1–6, doi: 10.1109/CSICC52343.2021.9420563.

[14] B. Hu, Q. Chen, and F. Zhu, "LCSTS: A large scale Chinese short text summarization dataset," 2015, *arXiv:1506.05865*.

[15] M. Humayoun, R. Nawab, M. Uzair, S. Aslam, and O. Farzand, "Urdu summary corpus," in *Proc. 10th Int. Conf. Language Resour. Eval.*, 2016, pp. 796–800. [Online]. Available: https://aclanthology.org/L16-1128

[16] T. Rahman, "Language policy and localization in Pakistan: Proposal for a paradigmatic shift," in *Proc. SCALLA Conf. Comput. Linguistics*, vol. 99, 2004, pp. 1–19.

[17] A. Naseer and S. Hussain, "Supervised word sense disambiguation for Urdu using Bayesian classification," Center Res. Urdu Lang. Process., Lahore, Pakistan, Tech. Rep., 2009.

[18] A. Daud, W. Khan, and D. Che, "Urdu language processing: A survey," *Artif. Intell. Rev.*, vol. 47, no. 3, pp. 279–311, Mar. 2017, doi: 10.1007/s10462-016-9482-x.

[19] A. Nawaz, M. Bakhtyar, J. Baber, I. Ullah, W. Noor, and A. Basit, "Extractive text summarization models for Urdu language," *Inf. Process. Manage.*, vol. 57, no. 6, Nov. 2020, Art. no. 102383, doi: 10.1016/j.ipm.2020.102383.

[20] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," 2015, *arXiv:1509.00685*.

[21] S. Chopra, M. Auli, and A. M. Rush, "Abstractive sentence summarization with attentive recurrent neural networks," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, 2016, pp. 93–98, doi: 10.18653/v1/n16-1012.

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

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

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

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

[26] J. Niu, M. Sun, J. J. P. C. Rodrigues, and X. Liu, "A novel attention mechanism considering decoder input for abstractive text summarization," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–7, doi: 10.1109/ICC.2019.8762040.

[27] F. Nan, R. Nallapati, Z. Wang, C. N. dos Santos, H. Zhu, D. Zhang, K. McKeown, and B. Xiang, "Entity-level factual consistency of abstractive text summarization," 2021, *arXiv:2102.09130*.

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

[29] Y. Zhao, Z. Luo, and A. Aizawa, "A language model based evaluator for sentence compression," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 170–175, doi: 10.18653/v1/p18-2028.

[30] K. Filippova, E. Alfonseca, C. A. Colmenares, L. Kaiser, and O. Vinyals, "Sentence compression by deletion with LSTMs," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 360–368, doi: 10.18653/v1/d15-1042.

[31] F. Liu, J. Flanigan, S. Thomson, N. Sadeh, and N. A. Smith, "Toward abstractive summarization using semantic representations," 2018, *arXiv:1805.10399*.

[32] Y. Dong, S. Wang, Z. Gan, Y. Cheng, J. C. K. Cheung, and J. Liu, "Multi-fact correction in abstractive text summarization," 2020, *arXiv:2010.02443*.

[33] Q. A. Al-Radaideh and D. Q. Bataineh, "A hybrid approach for Arabic text summarization using domain knowledge and genetic algorithms," *Cognit. Comput.*, vol. 10, no. 4, pp. 651–669, Aug. 2018, doi: 10.1007/s12559-018-9547-z.

[34] D. Suleiman and A. Awajan, "Multilayer encoder and single-layer decoder for abstractive Arabic text summarization," *Knowl.-Based Syst.*, vol. 237, Feb. 2022, Art. no. 107791, doi: 10.1016/j.knosys.2021.107791.

[35] B. Gunel, C. Zhu, M. Zeng, and X. Huang, "Mind the facts: Knowledge-boosted coherent abstractive text summarization," 2020, *arXiv:2006.15435*.

[36] N. Alexandr, O. Irina, K. Tatyana, K. Inessa, and P. Arina, "Fine-tuning GPT-3 for Russian text summarization," in *Proc. 5th Comput. Methods Syst. Softw. Data Sci. Intell. Syst.* Cham, Switzerland: Springer, 2021, pp. 748–757, doi: 10.1007/978-3-030-90321-3_61.

[37] T. Goyal, J. J. Li, and G. Durrett, "News summarization and evaluation in the era of GPT-3," 2022, *arXiv:2209.12356*.

[38] I. Gusev, "Dataset for automatic summarization of Russian news," in *Proc. 9th Conf. Artif. Intell. Natural Lang.*, Springer, 2020, pp. 122–134, doi: 10.1007/978-3-030-59082-6_9.

[39] K. Akiyama, A. Tamura, and T. Ninomiya, "Hie-BART: Document summarization with hierarchical BART," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Student Res. Workshop*, 2021, pp. 159–165, doi: 10.18653/v1/2021.naacl-srw.20.

[40] M. La Quatra and L. Cagliero, "BART-IT: An efficient sequence-to-sequence model for Italian text summarization," *Future Internet*, vol. 15, no. 1, p. 15, Dec. 2022, doi: 10.3390/fi15010015.

[41] K. Hussain, N. Mughal, I. Ali, S. Hassan, and S. M. Daudpota, (2021), "Urdu news dataset 1M," *Mendeley Data*, doi: 10.17632/834vsxnb99.3.

[42] Y. Cui, S. Wang, and J. Li, "LSTM neural reordering feature for statistical machine translation," 2015, *arXiv:1512.00177*.

[43] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*.

[44] Y. Chu, X. Yue, L. Yu, M. Sergei, and Z. Wang, "Automatic image captioning based on ResNet50 and LSTM with soft attention," *Wireless Commun. Mobile Comput.*, vol. 2020, pp. 1–7, Oct. 2020, doi: 10.1155/2020/8909458.

[45] M. Patel, A. Chokshi, S. Vyas, and K. Maurya, "Machine learning approach for automatic text summarization using neural networks," *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 7, no. 1, pp. 194–202, 2018.

[46] S. Song, H. Huang, and T. Ruan, "Abstractive text summarization using LSTM-CNN based deep learning," *Multimedia Tools Appl.*, vol. 78, no. 1, pp. 857–875, Jan. 2019, doi: 10.1007/s11042-018-5749-3.

[47] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 3104–3112.

[48] S. Haider, "Urdu word embeddings," in *Proc. LREC*, 2018, pp. 964–968. [Online]. Available: https://aclanthology.org/L18-1155.pdf

[49] M. N. A. Ali and G. Tan, "Bidirectional encoder–decoder model for Arabic named entity recognition," *Arabian J. Sci. Eng.*, vol. 44, no. 11, pp. 9693–9701, Nov. 2019, doi: 10.1007/s13369-019-04068-2.

[50] K. Al-Sabahi, Z. Zuping, and Y. Kang, "Bidirectional attentional encoder–decoder model and bidirectional beam search for abstractive summarization," 2018, *arXiv:1809.06662*.

[51] E. Jobson and A. Gutiérrez, "Abstractive text summarization using attentive sequence-to-sequence RNNs," 2016, p. 8.

[52] K. Lopyrev, "Generating news headlines with recurrent neural networks," 2015, *arXiv:1512.01712*.

[53] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," 2019, *arXiv:1910.13461*.

[54] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.
[55] T. B. Brown et al., "Language models are few-shot learners," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 1877–1901.
[56] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, p. 9, 2019.
[57] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané, "Concrete problems in AI safety," 2016, *arXiv:1606.06565*.
[58] A. M. Lamb, A. G. A. P. Goyal, Y. Zhang, S. Zhang, A. C. Courville, and Y. Bengio, "Professor forcing: A new algorithm for training recurrent networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4601–4609.
[59] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017, *arXiv:1711.05101*.

MUHAMMAD AWAIS received the B.S. degree in electronics and communication from The University of Lahore, Lahore, in 2012, and the M.S. degree in electrical engineering from COMSATS University Islamabad, Islamabad, Pakistan, in 2015. He is currently pursuing the Ph.D. degree in computer science with COMSATS University Islamabad, Lahore. He is an Assistant Professor with the Faculty of Engineering and Technology, The University of Lahore. He has published more than ten research publications in well-reputed international journals and conferences. His research interests include natural language processing, sequence-to-sequence deep learning models, large language models, transformers, and text summarization.

RAO MUHAMMAD ADEEL NAWAB received the Ph.D. degree in computer science from The University of Sheffield, U.K. He is currently an Assistant Professor with the Computer Science Department, COMSATS University Islamabad, Lahore Campus, Pakistan. His research interests include natural language processing, text reuse, plagiarism detection, author profiling, word sense disambiguation, text summarization, image captioning, information retrieval, question-answering systems, and paraphrase detection.