scientific reports

OPEN

Paraphrase detection for Urdu language text using fine-tune BiLSTM framework

Muhammad Ali Aslam[1], Khairullah Khan[1], Wahab Khan[1], Sajid Ullah Khan[2]✉, Abdullah Albanyan[3] & Shabbab Ali Algamdi[4]

Automated paraphrase detection is crucial for natural language processing (NL) applications like text summarization, plagiarism detection, and question-answering systems. Detecting paraphrases in Urdu text remains challenging due to the language's complex morphology, distinctive script, and lack of resources such as labelled datasets, pre-trained models, and tailored NLP tools. This research proposes a novel bidirectional long short-term memory (BiLSTM) framework to address Urdu paraphrase detection's intricacies. Our approach employs word embeddings and text preprocessing techniques like tokenization, stop-word removal, and label encoding to effectively handle Urdu's morphological variations. The BiLSTM network sequentially processes the input, leveraging both forward and backward contextual information to encode the complex syntactic and semantic patterns inherent in Urdu text. An essential contribution of this work is the creation of a large-scale Urdu Paraphrased Corpus (UPC) comprising 400,000 potential sentence pair duplicates, with 150,000 pairs manually identified as paraphrases. Our findings reveal a significant improvement in paraphrase detection performance compared to existing methods. We provide insights into the underlying linguistic features and patterns that contribute to the robustness of our framework. This resource facilitates training and evaluating Urdu paraphrase detection models. Experimental evaluations on the custom UPC dataset demonstrate our BiLSTM model's superiority, achieving 94.14% accuracy and outperforming state-of-the-art methods like CNN (83.43%) and LSTM (88.09%). Our model attains an impressive 95.34% accuracy on the benchmark Quora dataset. Furthermore, we incorporate a comprehensive linguistic rule engine to handle exceptional cases during paraphrase analysis, ensuring robust performance across diverse contexts.

Keywords Paraphrase detection, BiLSTM, NLP, LSTM, CNN, Urdu text

Researchers in several domains, such as information retrieval[1], plagiarism detection[2], and machine translation[3], have shown interest in automated paraphrase identification. Among the benefits of automating this procedure are the enhancement and Optimization of natural language document classification, sorting, and extra analysis. There are some difficulties with paraphrase detection[4]. These difficulties include handling typos, filler words, and jargon, coping with homonymy and polysemy problems, and handling situations where compared phrases lack lexical overlap[5]. The exact words or phrases can have various meanings in different circumstances in natural language, even when used in other phrases to communicate comparable ideas. According to the study conducted in[6], paraphrasing is a linguistic method frequently used in text reuse and plagiarism. The text must be changed while retaining its original meaning. Various definitions of paraphrasing include the following: deleting extra contexts resulting from syntactic changes; using synonyms for words; structural changes including the swapping of words, flipping between active and passive sentences; and summarizing[7].

The study presents the development of a new bi-directional LSTM based deep learning model specially developed for Urdu paraphrase detection. Previous approaches, including n-gram models, Siamese networks, and transformer-based architectures, have been used in English and for other languages, but translated directly to Urdu and presents a problem because of the language's morphology, script, and lack of available resources[8].

[1]Department of Computer Science, University of Science and Technology, Bannu 28100, Pakistan. [2]Department of Information Systems, College of Computer Engineering and Sciences, Prince Sattam Bin Abdul Aziz University, Al-Kharj, Kingdom of Saudi Arabia. [3]Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam Bin Abdul Aziz University, Al-Kharj, Kingdom of Saudi Arabia. [4]Department of Software Engineering, College of Computer Engineering and Sciences, Prince Sattam Bin Abdul Aziz University, Al-Kharj, Kingdom of Saudi Arabia. ✉email: sk.khan@psau.edu.sa

Our approach takes advantage of using BiLSTM networks in modeling dependencies and contextual features, which is especially important for considering the complex structure of the Urdu language. In response to the shortage of data annotated for this task, we have developed the UPC, a vast collection of Urdu paraphrase pairs identified by hand. Therefore, in order to adapt the BiLSTM model for Urdu paraphrase identification for which there is currently little to no resource available, this method will be fine-tuned on this custom-built dataset.

There already exist approaches of paraphrase detection concerning the entire paragraph, in our paper we address the problem of paraphrase detection at the sentence level for the Urdu language. This decision is based on the following reasons. Firstly, sentence-level paraphrase detection is amongst the requisites for a complex tier of tasks like plagiarism, text summery and question and answer systems. Meeting this core task allows for plundering the future extensions for further work on longer texts and complex language structures[9]. Secondly, the difficulties that Urdu language present such as its morphological complexity, script, its scarcity, are prominent at the sentence level and therefore it is preferred as the starting point to build good NLP systems[10]. Forming comprehensive original datasets with important sentences paraphrastic annotations is challenging as it requires manual work[11]. Thus, targeting this task, we provide a helpful resource for the Urdu NLP community and lay the foundation for exploring further extended tasks, including the paragraph-level paraphrase detection.

The identification of paraphrase in Urdu calls for distinctiveness essentially due to its structural system, script and language peculiarities. For this reason, we integrate a set of rule as high-level language processing rules based on detailed investigation of Urdu language characteristics. This is the rule-based system that formalities the treatment of abnormal and special conditions, constructional irregularities, and other variants likely to be found during paraphrase study and analysis, to enhance robustness. In the previous work, the n-gram based approaches and simple features like the bag-of-words, were used as the features for Urdu paraphrase detection and were not found to be very effective. The underlying objectives of these techniques largely depend on lexical match, which makes them irrelevant when paraphrase bear low lexical similarity but maps to the same meaning after structural-semantic transformations[12].

In corrections, the improvement in the detection of paraphrase has been witnessed particularly by deep learning models such as CNNs and LSTM networks as recent interventions in the learning field show[13]. These models are yet to capture long range dependencies and contexts which are necessarily present in the Urdu text and they fail to identify paraphrase when there exist other syntactic phenomena[14]. This work investigates several challenges in developing the Urdu paraphrase detection the lack of resources is illustrated by the following: The availability of annotated paraphrase dataset The difficult morphology of Urdu language The distinct writing system[15]. Some of the paraphrase detection methods developed, evaluated and compare are based on these benchmark datasets. It has also helped them increase linguistic knowledge of paraphrasing. Nevertheless, considering the lack of benchmark datasets and the overall lack of NLP work in these languages, there is an urgent need to shift focus toward resource-scarce languages, specifically Indo-Aryan languages[16]. The detection of paraphrases has previously been accomplished using a range of machine learning and deep learning techniques, such as N-gram-based approaches[17], LSTM approaches[18], and deep learning-based word embedding techniques[19]. It is crucial to remember that methods that work well on one dataset might not translate well to another corpus[20]. Syntax, semantics, and structure variations between European languages, Indo-Aryan languages, and English hamper the straightforward application of existing methodologies.

Consequently, particular corpora and techniques designed for low-resource Indo-Aryan languages are required. We suggest using a BiLSTM model for Urdu paraphrase detection to close this gap. Urdu is a widely spoken South Asian language but faces challenges due to the lack of NLP tools and research corpora. Urdu writing style is done using the Nastaleeq script that is write from right to left. It has borrowed most of its words from Arabic, Turkic and Persian languages[21]. English along with Arabic is written in Unicode text and often for Urdu and Arabic text specific technology and tools are needed as a result the proper methodology has to be used as these languages use different alphabet and characters as compare to English. This we do by utilising the BiLSTM model which provides hardware for long term storage and bidirectionality for processing sequential data.

To overcome these limitations, the following new bidirectional long short-term memory (BiLSTM) model is developed featuring Urdu paraphrase identification. Our approach uses the benefits offered by BiLSTM networks that unites both forward and backward information contexts to provide a precise encoding of specific syntactic and semantic connections amazing in Urdu language. Handling morphological complexities and exceptions of Urdu language during paraphrasing is efficiently addressed in our model by using word embeddings, text preprocessing methodologies and integrated high-end rule based linguistic engine.

One distinctive feature of this work is the construction of a large-scale Urdu Paraphrased Corpus (UPC) which consists of 400,000 potential sentence pair duplicate and 150,000 are manually endorsed as paraphrase. This valuable dataset is particularly useful to the identified research gap of the lack of labelled data for Urdu and the need for models trained specifically for the language.

Through extensive experimental evaluations on the custom UPC dataset and the benchmark Quora dataset, we demonstrate the superior performance of our BiLSTM framework, outperforming state-of-the-art methods and achieving impressive accuracy scores. A meticulously designed linguistic rule engine further enhances the model's robustness, enabling accurate paraphrase detection even in irregular or edge cases.

Contributions

This work contributes a domain-specificBiLSTM model and a high-quality paraphrase corpus to the progress of Urdu NLP research. The difficulty of Urdu-language paraphrase recognition has not gotten much attention from NLP researchers until now. This effort aims to address that. The following are this work's main contributions:

- For paraphrase detection, gather information from various sources, including books, blogs, social media, news stories, and other written content, to create a fresh Urdu text corpus.
- Preprocess Urdu texts using various techniques, including text cleaning, tokenization, stop word removal, noise removal, word embeddings, and label encoding, to train a BiLSTM network for paraphrase recognition.
- The creation and application of the BiLSTM model, which was specially created to handle the challenges of paraphrase identification in Urdu, constitutes the study's primary contribution. The BiLSTM network effectively captures Urdu's intricate sentence patterns and linguistic complexities, enabling the accurate identification of paraphrases.
- The input network sequentially analyses the BiLSTM text, considering the word and phrase sequence. Because word order significantly affects meaning, sequential processing is especially well-suited for natural language processing (NLP) tasks like paraphrase detection.

Paper organization

The remaining paper is organized as follows: the literature review is discussed in Sect. 2, along with its significant contribution. The proposed technique and network design are described in Sect. 3. The experiments used to test the proposed model and produce the results are conducted in Sect. 4. The results and comparison with other approaches are presented and discussed in Sect. 5. Finally, Sect. 6 concludes this work and provides suggestions for future works.

Literature review

The Urdu language presents unique challenges for natural language processing due to its morphological complexity, syntactic richness, and scarcity of annotated datasets. These challenges make tasks such as paraphrase detection highly demanding and emphasize the need for tailored approaches.The study in[22] proposed a novel method that blends deep learning and the text similarity methodology to improve paraphrase detection. To obtain the semantic vector representation of each sentence, the researchers used a skip-thought deep learning model. Various similarity measures were used to determine how similar the generated semantic vectors were to one another. To capture the semantic links between words in a phrase[23], suggested two variants of the tree-structured LSTM model. Based on the semantic construction of the sentences, these models evaluated the similarity between the two sentences. An Accuracy score measured the degree of resemblance between the sentences. The available literature shows that the lack of specialized corpora makes it challenging to research and create paraphrased plagiarism detection algorithms for less-resourced languages like Urdu. The Urdu Paraphrase Plagiarism Corpus (UPPC) was created by a study conducted by[24] and is one of the few manually generated gold-standard corpora for Urdu. **T**hese resources primarily concentrate on document-paraphrased plagiarism.

Paraphrase detection is a challenging task that assists in finding sentences that convey similar meanings[25]. The problem of paraphrase detection has been researched using bi-encoder structures. The study[26] used LSTM in a twin architecture with coupled weights and the Manhattan distance to determine similarity. The authors of[27] also used a twin structure with a fullyconnected layer and BiLSTM for paraphrase detection. Different techniques have been used to extract features and find reuse instances in monolingual and multilingual documents. Some approaches use lexical matching to determine how similar two text segments are based on the number of phrases they share. The ability of these measures to capture semantic similarity above a fundamental level is constrained. Various word-embedding-based models, especially those that employ Word2Vec, GloVe (Global Vectors for Word Representation), and FastText, have shown competitive results in encoding contextual semantic meaning[28]. The study[29] suggested mathematical functions for unsupervised paraphrase detection, emphasizing the creation of asymmetrical paraphrase corpora. Deep learning techniques, particularly Siamese designs for neural networks[30], have gained significant attention in paraphrase detection. Siamese networks are dual-branch networks combined using an energy function and share the exact weights, allowing them to detect distinctions between input samples. The Manhattan distance function or cosine similarity function is frequently used to combine the outputs of Siamese LSTM models[31], in which each input text is fed into an LSTM sequence. The development of BERT, which revolutionized the field of NLP and achieved outstanding results in various GLUE tasks, including paraphrase detection, was made possible by the attention-based transformer model[32].

Similarly, a study presented in[33] investigated the available datasets for citation intent and proposed an automated citation intent technique to label the citation context with citation intent. Furthermore, the authors annotated ten million citation contexts with citation intent from the Citation Context Dataset (C2D) dataset with the help of our proposed method. Global Vectors (GloVe) were employed in previous studies to extract word-level semantic features, demonstrating strong performance but with notable limitations in capturing contextual nuances.Infersent, and Bidirectional Encoder Representations from Transformers (BERT) word embedding methods and compared their Precision, Recall, and F1 measures. BERT embedding performed significantly better, having an 89% Precision score. Another study was motivated by the accomplishments of deep learning algorithms and word embeddings in English sentiment analysis. Extensive experiments were conducted based on supervised machine learning in which word embeddings were exploited to determine the sentiment of Arabic reviews. Three deep learning algorithms were introduced: convolutional neural networks (CNNs), long short-term memory (LSTM), and a hybrid CNN-LSTM. The models used features learned by word embeddings such as Word2Vec and fastText rather than handcrafted features. The models were tested using two benchmark Arabic datasets: Hotel Arabic Reviews Dataset (HARD) for hotel reviews and Large-Scale Arabic Book Reviews (LARB) for book reviews, with different setups. Comparing experiments adopted the three models with two-word emeddings and others on the configs of the datasets[34]. Compared to the benchmark HARD dataset, the proposed CNN model achieved the highest accuracy of 94.69%, 94.63% and 94.54% for fast text, outcompeting the LSTM and CNN-LSTM models. In[35], the authors present a MPAN which is a deep learning-based multilevel parallel attention neural model that uses PBES to compute contextualized embeddings at character, word, and sentence level at the same time. The MPAN model then calculates multiple attention vectors at the multilevel and appends them to give the output: competitive accuracy. Concretely, the proposed MPAN model achieves ASR performance comparable to the state of the art, establishing new ASR baselines for 34 publicly accessible ASA databases. The proposed model is further shown to create new state-of-the-art accuracies for two multidomain collections: 95.To a binary classification collection, 61% average was achieved while to a tertiary classification collection, 94.25% average was achieved.

CNN and especially RNN, and among them LSTM networks, have been thoroughly researched in various NLP areas including paraphrase identification. CNNs have shown very high accuracy in learning local patterns and have been shown to take out n-gram features from the text. In the same respect, LSTMs have been beneficial in modeling sequences of data and forward dependencies that are profitable when handled with text sequences. The same can also be used in NLP tasks through adversarial training and generation of synthetic data using Generative Adversarial Networks (GANs). Blockchain technology and its concept of decentralism, transparency, and other features focus on the secure protection of data in the NLP system[36]. With the distributed architecture of the ledger, blockchain provides great safety against data theft and hacking. Spatial attention mechanisms that capture the interactions of the spatial regions in the data enhance several computer vision processes and can be adopted for NLP tasks that entail geometric data or vision-question-answering tasks[37].

Updated developments in GAN and generative models including Transformer based architectures[38] have been used in production of contextually coherent and relevant images and textual formats. These models can in turn be extended to provide paraphrase generation tasks and use the results to generate multiple paraphrases of a given input text. Research has been conducted regarding the use of other deep recurrent networks, including Bidirectional LSTMs (BiLSTMs) that have been applied to learning of paraphrasing in text[39] owing to its capacity of considering contextual information in both directions of a given input sequence.

Prior research on Urdu NLP has mainly targeted on various NLP tasks like text classification, sentiment analysis and transliteration. These studies show that Ur - du deep characteristics which include its script, word formation and semantics that are rather complex. It observed that conventional writing styles, such as CNNs, have not been particularly effective for Urdu paraphrase detection as they are not capable of addressing contextual associations across long sequences. Although the LSTM models are slightly superior to other methods in handling sequential data, they have unidirectional drawbacks. For complex linguistic phenomena in Urdu, these are insufficient in certain ways which are described as follows: The proposed BiLSTM framework does not have these two limitations because it captures both forward and backward context while capturing the syntactic and semantic peculiarity of the Urdu language. It includes a linguistic rule engine for better case handling, which definitely increases the system's reliability compared to other similar approaches. Table X shows a way of comparison and differentiation of the models.

Some of the latest analysis has been conducted to establish how capsule networks could be used to enhance the execution of some rich NLP tasks such as paraphrase identification. Federated learning, a distributed machine learning approach enabling collaborative training while keeping data decentralized and privacy-preserving, has gained attention in NLP, allowing models to be trained on data across devices/organizations without centralizing sensitive information, aligning with data security and privacy principles[40]. These architectures have shown promising results in modelling hierarchical relationships and capturing spatial and structural information in data, which could be beneficial for handling structural variations and long-range dependencies in paraphrase detection.

By establishing stronger connections with the latest trends and advancements in NLP research, our work can be better contextualized and positioned within the broader research landscape. Exploring these emerging techniques and approaches could further improve and extend our proposed paraphrase detection framework.

Few attempts have been made to detect sentence-level paraphrases in Urdu due to the lack of corpus that concentrates on Urdu at the sentence level. With the advent of automated approaches for paraphrase detection, finding this sort of plagiarism remains challenging due to the heterogeneous nature of the databases on which these algorithms are trained[41]. As mentioned in the literature, minimal work has been conducted because there is a dearth of Urdu NLP resources, especially for paraphrase detection. Therefore, we intend to provide a novel Urdu paraphrase detection corpus. To evaluate the effectiveness of our proposed BiLSTM framework for Urdu paraphrase detection, we conducted extensive experiments employing two distinct datasets: a custom-built corpus tailored for the Urdu language and the widely-used Quora dataset. The custom dataset, meticulously curated from multiple Urdu language sources, comprises a substantial collection of sentence pairs classified as either paraphrases or non-paraphrases. Rigorous preprocessing techniques were applied to this dataset to ensure data quality and remove extraneous content. Conversely, the Quora dataset, originating from the popular question-and-answer platform, consists of sentence pairs labelled as paraphrases or non-paraphrases, allowing us to assess the algorithm's generalization capabilities on real-world data.

Research gap and questions

While the existing literature provides valuable insights into paraphrase detection techniques, several gapsmust be addressed, particularly for resource-constrained languages like Urdu. Most prior works have focused on European and English languages, leveraging extensive annotated datasets and pre-trained language models. Urdu has a scarcity of labelled paraphrase corpora, pre-trained models, and NLP tools tailored for its complex morphology and unique script. This lack of resources poses significant challenges in developing robust paraphrase detection systems for Urdu text.To bridge this gap, our research aims to address the following key questions:

- How can we create a high-quality, annotated corpus of Urdu paraphrase pairs to facilitate model training and evaluation?

- What deep learning architecture is best suited to capture the intricate linguistic patterns and contextual dependencies present in Urdu text for accurate paraphrase detection?
- Can we develop an efficient paraphrase detection system that outperforms existing techniques while addressing the resource constraints of the Urdu language?

By addressing these research questions, our work aims to advance the state-of-the-art in Urdu paraphrase detection and contribute to the broader development of NLP capabilities for this language.

Proposed framework

This work brings creation of the dataset being proposed as a key contribution of the work since it seeks to enhance Urdu NLP through accurate paraphrasing identification. The proposed Urdu paraphrase detection model uses BiLSTM altogether with new large scale data set construction. This dataset was compiled after gathering data from various resources in the Urdu language constituting of carefully labeled English sentence pairs as paraphrases or otherwise. Annotation of the data was carried out by five volunteers who are native Urdu speakers and confidentiality of the patients' information was maintained in the process Annotation of the samples was done independently and each sample was assigned to two different annotators for reliability. Concerning the formulation of different opinions, the consensus was reached by discussion, and among inter-annotator agreement scores ICC (2) was estimated to ensure the accuracy of annotations. Due to this, very effective preprocessing methods were used to clean up the data by eliminating any additional irrelevant information. Thus, the new dataset, based on the Full Quora contextual information, was employed in the training, as well as testing, and evaluating of the BiLSTM model. Thus, the emphasis on constructing a cleanly integrated dataset, including elements of annotation, underscores its significance as a crucial effort to this research. Such a dataset not only helps to assess the proposed model, but can also become a source for further developing this line of work in Urdu paraphrase detection. By overcoming the mentionedissues related to scarcity of data in LRL, this contribution pretty well enhances the repository of knowledge in Urdu NLP.

The more recent deep learning methodologies including the Transformer based models and improved neural architecture have attracted comparatively more attention to a large number of NLP tasks; the decision to use the BiLSTM paradigm for Urdu paraphrase identification was influenced by several reasons. First of all, BiLSTM is a highly effective architecture for paraphrasing in-sequence task, which require the utilization of contextual information and long-distance relationships. Secondly, there is the basic problem of working in Urdu, a language that has few large-scale annotated datasets available for machine learning, and even fewer pre-trained language models on which to base more state-of-the-art approaches. Accordingly, the proposed BiLSTM model is more suitable for this task because it provides a highly stable performance even with relatively more minor datasets than other models. The interpretability allowed by the system or the BiLSTM unit and the applicability of fine-tuning to the given paraphrase detection task are additional points in the favor of the presented approach to language modeling in the context of Urdu. While we acknowledge the potential benefits of more recent techniques, we focus on developing an effective and efficient solution tailored to the specific challenges of Urdu paraphrase detection, considering the trade-offs between model complexity, resource requirements, and performance.Rule-based approaches play a pivotal role in handling domain-specific nuances in NLP tasks. Kang et al.[42] demonstrated how rule-based processing improved disease normalization in biomedical texts. Similarly, Ledig et al.[43] highlighted the importance of preprocessing in enhancing model accuracy. Recent works by Khan et al.[44,45] discuss combining rule-based systems with deep learning techniques, particularly for low-resource languages, providing insights into designing robust frameworks.

Corpus creation process

The development and testing of paraphrase detection methods are severely hampered by a lack of resources, particularly for South Asian languages like Urdu. There is no sentence-level corpus of Urdu available yet. For resource-poor languages, corpus creation is needed. A novel Urdu language paraphrased corpus is developed in this work using a semi-automatic procedure based on the question pair technique from Quora[46]. The suggested corpus, the Urdu Paraphrased Corpus (UPC), is based on Quora but has been adjusted for Urdu text and contains sentences.

The Urdu Paraphrased Corpus (UPC) comprises 400,000 sentence pairs, of which 150,000 are annotated as paraphrases. Annotation was carried out by five native Urdu speakers skilled in paraphrasing. The paraphrases were generated using a combination of synonym substitution, sentence restructuring, and summarization techniques. A diverse range of sources books, blogs, news articles, and social media ensures linguistic richness and broad coverage of writing styles. Manual validation ensured semantic consistency and eliminated redundancy. The dataset is structured for public access upon reasonable request, providing a valuable resource for future Urdu NLP research.The primary objective is to simplify the evaluation and comparison of cutting-edge monolingual paraphrase detection methods specifically established for Urdu. The corpus aims to mimic paraphrasing strategies students regularly employ in academic contexts. Since it accurately reflects the paraphrasing techniques typically employed by students, the researchers decided to generate example cases for the UPC using a similar method. The UPC's primary area of development is the journalism industry. This decision is affected by several factors. First, the journalism industry acted as a practical source for original and copied news items because newspapers are readily available online in electronic form. Second, while the newspaper industry offers pertinent examples, scholars have difficulty locating actual cases of paraphrasing and plagiarism due to confidentiality concerns.

Newspapers typically employ various paraphrasing techniques, such as sentence combining or splitting, synonym substitution, tense or sentence structure changes, and summarizing. These methods were essential in creating a comprehensive library that accurately reflects the many paraphrase strategies applied in duplicated publications. Five native Urdu speakers skilled at paraphrasing were asked to physically write essays to produce paraphrased and original documents. They had access to articles from online Urdu newspapers as their source material, and they were instructed to edit the text while maintaining its semantics by adding the appropriate synonyms and rearranging the sentences. Written agreements were made with the volunteers to ensure the public's access to the corpus. The participants were given learning resources to help them produce non-plagiarized documents, including books, journals, and internet sources on their re-searching subjects. Using journalism sources for this task was expressly prohibited. The UPC is deliberately crafted using these principles to make it a valuable tool for comparing different paraphrase detection systems. There are 300,087 sentence pairs in the UPC, of which 167,015 pairs have been labelled as paraphrases by human experts. The average length of the text is typically between 10 and 15 words per sentence, and each sentence has between 7 and 12 unique words. The assessment of semantic equivalence occurred between an Urdu source sentence, defined as U with word embeddings, $\mathrm{U} = [\mathrm{u}_1, \mathrm{u}_2, \ldots, \mathrm{u}_\mathrm{n}]$, and a subset of sentences with word embeddings as $\mathrm{V} = [\mathrm{v}_1, \mathrm{v}_2, \ldots, \mathrm{v}_\mathrm{n}]$. The goal of the investigation is to determine whether the meanings of the two statements are equivalent. The proposed model is displayed in Fig. 1.

Linguistic rule engine for exceptional case handling

We propose a meticulously designed linguistic rule engine to address the exceptional cases and irregularities inherent in Urdu's linguistic landscape. This rule-based system encapsulates a comprehensive set of rules derived from an extensive analysis of Urdu's morphological, syntactic, and semantic intricacies. Despite the irregularity and other factors which make them what can be regarded as incremental edge cases, the rule engine works perfectly within our proposed BiLSTM framework.

The linguistic rule engine comprises three primary components: After this, there are these three parts; (1) Morphological Rule Component, (2) Syntactic Rule Component, and (3) Semantic Rule Component. The Morphological Rule Component deals with affairs of Urdu morphological Rules that are irregular word formations, compound words, and morphological Greeks. The Syntactic Rule Component deals with the problems of word order, its variations, irregularity in the syntactic structures in Urdu. Last but not least, it is the Semantic Rule Component that concerns itself with erraticisms pertaining to semasiology or semantics, idiom and context sensitive meaning flip-flops and paradoxes.

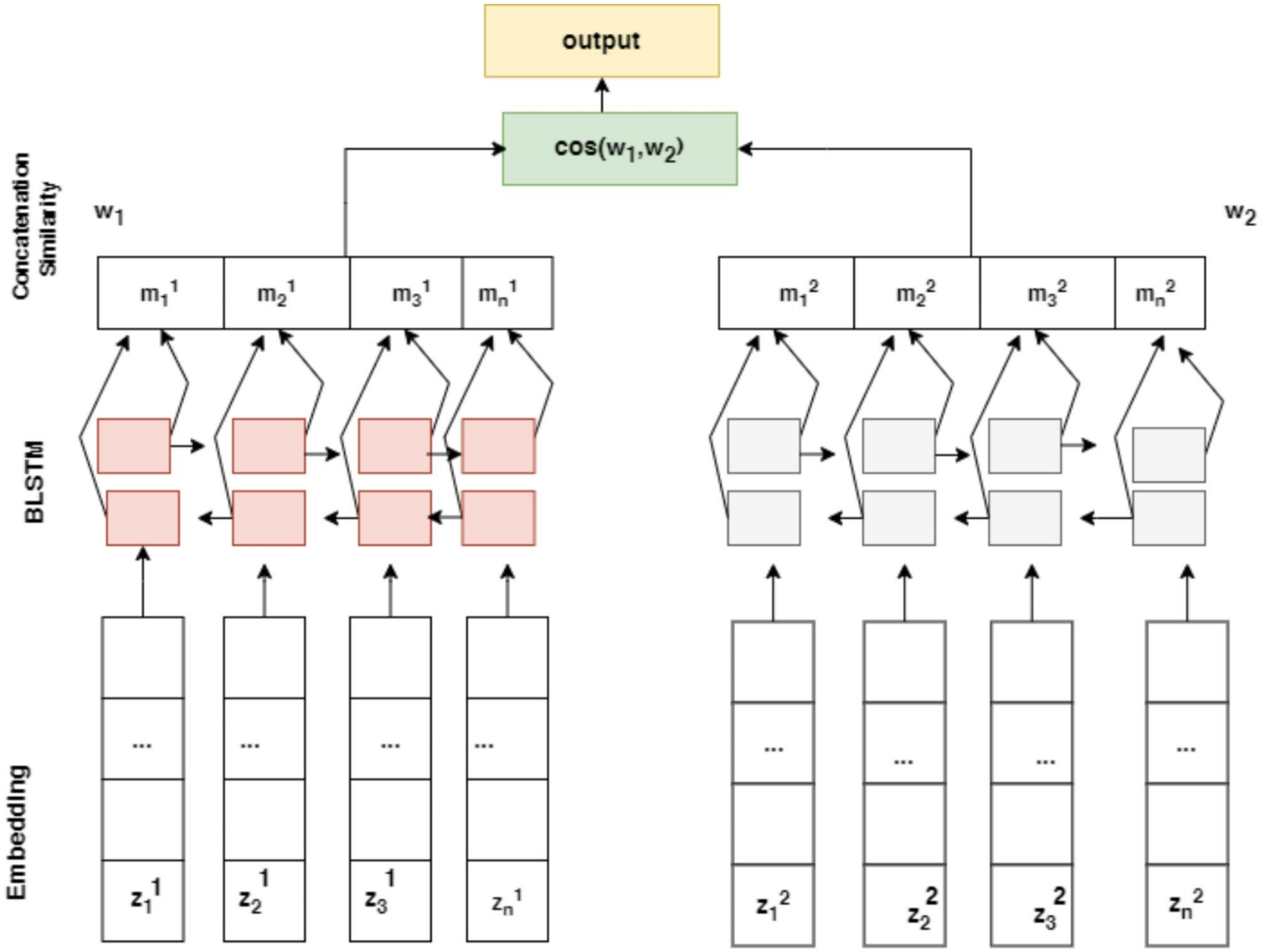

Fig. 1. Illustration of the proposed model for Urdu paraphrased detection.

By incorporating this extensive linguistic rule engine in our BiLSTM framework, it is made certain that our model is capable of identifying Urdu paraphrased text even though noises or anomalies that are not easily tractable in common setting do exist.

Word embedding

Using Word2Vec for training word embeddings proved effective in capturing the semantic relationships between words within a multi-dimensional space and leveraging their relatedness[47]. This study incorporates the GloVe approach for embedding words with 50 dimensions. In this study, we used Common Crawl pre-trained vectors trained on a large amount of web-based text (42 billion tokens, 1.9 million words, 50 d vectors). We downloaded the Common Crawl GloVe embeddings from the official GloVe website and used the Genism library. The foundation of GloVe is a global co-occurrence matrix in which each component, $\mathrm{Y_{ab}}$, represents the frequency with which terms $\mathrm{v_a}$ and $\mathrm{v_b}$ occur together in a specific context window. The GloVe model is trained using loss function L, described in Eq. (1).

$$\mathrm{L} = \sum_{\mathrm{a,b=1}}^{\mathrm{W}} \mathrm{f}\left(\mathrm{Y_{ab}}\right)\left(\mathrm{v_a^T}\hat{\mathrm{v}}\mathrm{b} + \mathrm{ca} + \mathrm{c_b} - \log\left(\mathrm{Y_{ab}}\right)\right)^2 \tag{1}$$

The embeddings of the words $\mathrm{v_a}$ and $\mathrm{v_b}$ are represented by $\mathrm{y_a}$ and $\mathrm{y_b}$, respectively, in the above Equation. The terms denote scalar biases linked with the corresponding words $\mathrm{c_a}$ and $\mathrm{c_b}$. When a word's frequency is too high, the weighting co-occurrence function $\mathrm{f_y}$ is employed. Here, V represents the vocabulary's overall length. This function lessens the impact of often occurring terms while ensuring a more realistic portrayal of word co-occurrence. Non-contextual embedding models like Word2Vec and GloVe, while effective in capturing semantic relationships based on word co-occurrence, face limitations in handling context-aware features and out-of-vocabulary (OOV) words. These embeddings provide static representations that fail to account for the dynamic nature of word meanings in different contexts, which is particularly critical for morphologically rich languages like Urdu. OOV words remain unrepresented, limiting the model's ability to generalize effectively. Future enhancements could explore integrating contextual embeddings, such as BERT or FastText, which dynamically generate context-sensitive word representations and mitigate the OOV issue by leveraging subword-level information.

LSTM model

Long-term dependencies in sequential data must be captured and learned using recurrent neural networks (RNNs), and the LSTM network is a popular model in this family. This network does text classification, regardless of whether it has been paraphrased or not, and for this purpose, three gates are used to maintain the memory state $\mathrm{C_t}$.: the input gate $\mathrm{i_t}$, the forget gate $\mathrm{f_t}$, and the output gate $\mathrm{o_t}$. By controlling the flows of contextual information, these gates facilitate the updating and retention of relevant information in the memory state. Furthermore, the gates cooperate to calculate the output and $\mathrm{h_t}$ of the hidden layer, which holds the aggregated information from the preceding layers. Making use of the input, output, and forget gates, the LSTM's cell state $\mathrm{C_t}$, which works as its long-term memory alters with time.

Input gate ($\mathrm{i_t}$): The input gate chooses which data from the present input $\mathrm{x_t}$ should be kept in the cell state. It accepts the previous hidden state $\mathrm{h_{t-1}}$ as well as the present input. Mathematically, it is described as:

$$\mathrm{i_t} = \sigma\left(\mathrm{W_{xi}x_t} + \mathrm{W_{hi}h_{t-1}} + \mathrm{b_i}\right) \tag{2}$$

The input gate uses input weight matrices $\mathrm{W_{xi}}$, input biases $\mathrm{b_i}$, and a decision-making mechanism to decide which data from the prior hidden layer $\mathrm{W_{hi}h_{t-1}}$ and the current input $\mathrm{x_t}$ should be kept in the cell.

Forget gate($\mathrm{f_t}$): The forget gate chooses which data from the previous cell state $\mathrm{C_{t-1}}$ should be forgotten. It receives data from the previous hidden state $\mathrm{h_{t-1}}$ as well as the current input state. It is calculated as:

$$\mathrm{f_t} = \sigma\left(\mathrm{W_{xf}x_t} + \mathrm{W_{hf}h_{t-1}} + \mathrm{b_f}\right) \tag{3}$$

The forget gate calculations assist in determining the significance and applicability of the data presently stored in the memory cell.

Cell state ($\mathrm{C_t}$): The cell state $\mathrm{C_t}$, is employed during sequential data processing to convey and retain long-term information. It is used to solve the vanishing gradient issue in the LSTM. It is calculated as:

$$\mathrm{C_t} = \mathrm{f_t} \odot \mathrm{C_{t-1}} + \mathrm{i_t} \odot \tilde{\mathrm{C}}_{\mathrm{t}} \tag{4}$$

Hidden state ($\mathrm{h_t}$): Following the LSTM cell processing, the hidden state is calculated by applying an activation function to the updated cell state. The output gate controls the part of the cell state used for calculating the hidden state.

$$\mathrm{h_t} = \mathrm{o_t} \odot \tanh\left(\mathrm{C_t}\right) \tag{5}$$

The final hidden state of the LSTM network conveys the semantic meaning of the phrase and contains its contextual information after analyzing the complete sequence.

Output gate ($\mathrm{o_t}$): The output gate chooses which data from the cell state $\mathrm{C_t}$ need to be output as the current hidden state $\mathrm{h_t}$, which serves as the LSTM's final result for the current time step. It is computed as:

$$o_t = \sigma\left(W_{xo}x_t + W_{ho}h_{t-1} + b_o\right) \tag{6}$$

Bi-LSTM framework

The LSTM model ignores subsequent information and only considers information that came before it in the input sequence. A BiLSTM architecture captures more data through contextual information to overcome this issue. The fundamental concept is to employ two LSTM models, one of which processes the sentences forward while the other processes them backward. As shown in Eq. (8), the outputs of these models are then concatenated to simplify additional processing. Using this method, the BiLSTM effectively captures the input sequence's past and future contextual dependencies. The architecture of the BiLSTM is taken from[48], as shown in Fig. 2.

The forward LSTM layer performs the input sequence processing in the forward direction, changing its hidden state $h_t^{(f)}$, as follows at each time step t.

$$h_t^{(f)} = \mathrm{LSTM}_{\mathrm{forward}}\left(x_t, h_{t-1}^{(f)}\right) \tag{7}$$

Where the LSTM cell function is called $\mathrm{LSTM}_{\mathrm{forward}}$ calculates the forward hidden state utilizing the current input x_t and the previous forward hidden state $h_{t-1}^{(f)}$.

To process the input sequences x_t in the backward direction, the backward LSTM updates its hidden state $h_{t-1}^{(b)}$ as follows at each time step t:

$$h_t^{(b)} = \mathrm{LSTM}_{\mathrm{backward}}\left(x_t, h_{t+1}^{(b)}\right) \tag{8}$$

The combination of the forward $h_t^{(f)}$ and backward $h_t^{(b)}$ hidden states arethe final result of the BiLSTM model at each time step t:

$$y_t = \left[h_t^{(f)}, h_t^{(b)}\right] \tag{9}$$

For a range of sequence modelling applications, the model's enhanced understanding of the dependencies and connections within the sequence becomes beneficial.

Experiments

To assess the performance of the designed BiLSTM framework for the paraphrase identification task of Urdu text, we perform a series of experiments. First, we created a novel dataset skilled towards Urdu noise to have refined and adjusted the paraphrase acknowledgment method. Besides, in an attempt to determine the general capabilities of the model, we tested the BiLSTM paraphrase detection on Quora dataset which is among the most utilized datasets to measure real-world data performance. Quora dataset is based on question and answer website of Quora, where we extracted sentence pairs as paraphrases or non-paraphrases just like in our Urdu

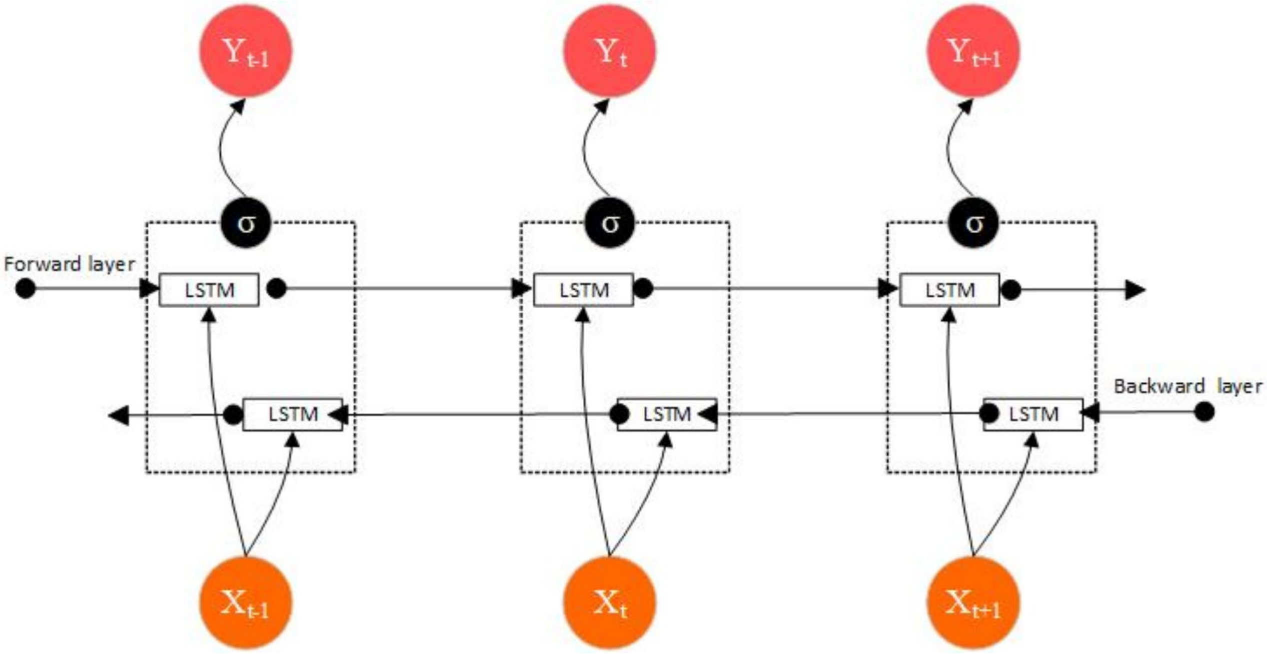

Fig. 2. BiLSTM architecture used in this work[48].

dataset. In order to assess how the proposed BiLSTM model works, we proceed with experiments on various baselines, which are CNN, LSTM, CNN-LSTM, and BERT, respectively, so as to give a fuller explanation for the accomplishment of the study.

Baseline models

CNN

CNNs are typically employed for image processing, but by considering sequential data, like text, as a one-dimensional signal, they can be applied to this data type. Our objective in paraphrase detection is to identify whether two input statements or phrases communicate the same meaning despite variations in phrasing or structure.

Local patterns and characteristics in the input data are very well-captured by CNNs. Local patterns are essential for locating comparable phrases or sentence fragments in the case of paraphrase identification, even when the overall structure is different. The parameters of CNN used for paraphrase detection are displayed in Table 1.

LSTM

The ability of LSTM to successfully capture long-range relationships and contextual data makes it appropriate for locating paraphrases in Urdu text. The text in Urdu is composed of a series of words, and LSTM was created primarily to handle sequential data. Sentences with the hidden states can be word-by-word processed to provide an ability of temporal correlation recording of the LSTM, which is necessary for the identification of the given sentence context and its meaning. It employs gates to regulate information, this differentiates between euphemisms that are very similar in construction but have almost the same meaning in contemporary English.

CNN-LSTM

This work employed Urdu English literature to classify the paraphrases using the CNN-LSTM. This model incorporates the feature extraction capability of CNN with the LSTM for narrow textual patterns recognition. It analyzes input text using convolutional layers to obtain n-gram features. Max-pooling is used to reduce dimensionality and locate significant characteristics. The LSTM layer retains long-term dependencies and information within feature maps. Fully connected layers retrieve higher-level features and conduct a binary categorization, the output layer deciding whether or not the input sentences are paraphrases.

BERT

This study also employed the BERT (Bidirectional Encoder Representations from Transformers) model for paraphrase identification. It acquires contextual embedding by guessing missing words and comprehending word connections in different situations. The BERT model is fine-tuned throughout the training phase using a labelled dataset of pairs of sentences classified as either paraphrases or non-paraphrases. The parameters of the model are modified using gradient descent and backpropagation.

Model evaluation criteria

In this research, we evaluated the effectiveness of our proposed model by using accuracy. The percentage of successfully detected paraphrase pairings in the assessment dataset divided by the total number of pairs represents the accuracy of paraphrase detection. It can be calculated using Eq. (10).

$$\mathrm{Accuracy} = \frac{\mathrm{T_P} + \mathrm{T_N}}{\mathrm{T_P} + \mathrm{T_N} + \mathrm{F_P} + \mathrm{F_N}} \tag{10}$$

Where $\mathrm{T_P}$ indicates true positive values, while $\mathrm{T_N}$represents true negative values. Furthermore, $\mathrm{F_P}$ and $\mathrm{F_N}$ indicate false positive and false negative values, respectively. The F-1 score is calculated as:

$$\mathrm{F} - 1 = \frac{\mathrm{T_P}}{\mathrm{T_P} + 1/2(\mathrm{F_P} + \mathrm{F_N})} \tag{11}$$

CNN and LSTM models, while effective for feature extraction and sequential data processing, exhibit limitations in capturing word- and sentence-level semantic nuances. These models lack mechanisms to focus on crucial word-level features dynamically, which is critical for semantic alignment in complex languages like Urdu. Attention mechanisms, which prioritize relevant parts of the input, can significantly enhance the capability of these models by improving focus on important features. Future work could integrate attention-based architectures or hybrid

Parameters	Description
Input text	Glove words embedding, 50-dimension
Convolutional layers, 3, activation function	Capture n-gram features, ReLU.
Pooling layer, 1	Max-pooling
Fully connected layer, 1	Higher-level feature extraction
Output layer, activation function	Final classification, sigmoid

Table 1. Parameters of CNN used for paraphrase detection.

models, such as attention-enabled LSTMs or transformer-based systems, to address these limitations effectively and achieve better semantic comprehension.

Environment setting

We utilized the environment parameters depicted in Table 2 to conduct experiments on the proposed paraphrase detection task using CNN, LSTM, CNN-LSTM, BERT, and BiLSTM networks. To optimize hyperparameters and model architectures, we conducted several experiments for paraphrase detection tasks using both Quora and a custom-built dataset. Cross-validation, early stopping criteria, and hyperparameter adjustment were performed to improve the model's accuracy further. TensorBoard was leveraged to enhance comparability and experiment tracking.

To ensure reproducibility, we provide the details of our experimental setup. The BiLSTM model was trained using the Adam optimizer with a learning rate of 0.001. A batch size of 32 was used for all experiments, and a dropout rate of 0.3 was applied to prevent overfitting. The training-validation-testing split was set at 80%−10%−10%, ensuring a representative evaluation of the model. Experiments were conducted on a system with an NVIDIA GTX 1060 GPU, 16 GB RAM, and PyTorch as the deep learning framework.

Addressing overfitting

The main problem associated with overfitting is that it is sometimes achieved at the cost of the models ability to generalize on unknown testing data sets. Overfitting is always a challenge in any training model and in the case of BiLSTM that we used for Urdu paraphrase detection the following strategies were adopted.

First, we applied early stopping throughout the training process. These features we could look at a model's cross-validation score set, where one could see exactly when the model began to over-fit the training data. This training stopped at the mentioned stage so that the model could not memorize the training data while boosting the generalization capability.

To avoid overfitting, we used two methods that are dropout and L 2 regularization, we also used validation loss to check the performance of the model. During the training session, Dropout randomly sets a fraction of neurons within a layer to zero and hence discourages the DNN from over depending on one or many neurons. L2 regularization also named weight decay, add a term to the loss function that prevents based on weights values in order to reduce them and have a balanced values.

To do this, we aimed at guaranteeing that the training data set was rich in the required diversity and similar to the target domain. To this end, we collected a large number of Urdu sentence pairs from books, blogs, social media accounts, news articles, and other written content, where the model was trained on a variety of linguistic routines and fluctuations. A group of models was more beneficial in this decision since it prevented overfitting of particular writing styles or domains.

To check the efficacy of the mitigation strategies we have proposed above, we kept a regular check on the accuracy of the model on both the training and the validation set while training the model. A small difference was observed between the training/ validation sets which confirmed that they are not overfitting on the training data and can perform well on new samples.

By applying these techniques, we were able to avoid high degrees of overfitting in the BiLSTM framework for Urdu paraphrase detection that we discussed above and gain good levels of generalization across data sets. As it bears features which help it to learn the data dependency feature, LSTM model has its weaknesses most of which stem from the question of vanishing gradients especially when dealing with long sequences. This constrained its capacity for maintaining such long-range dependencies of considerable importance. LSTMs are very sensitive to overfitting and may produce poor results especially when trained on small or unbalanced datasets. To address these challenges, this study used the following approaches which include; gradient clipping and regularization techniques, dropout, and early stopping. Future work might investigate using stronger architectures which are BiLSTMs with attention or transformers to mitigate these challenges effectively.

Parameter	Description
Operating system, CPU	Windows 10 with 64-bit architecture, Intel Core i7
RAM, GPU	16 GB, NVIDIA CUDA (GTX 1060)
Deep learning framework	Pythorch was used for experiments
Dataset	Quora dataset and custom develop dataset
Word embeddings	GloVe model
Model_Input	Word embeddings
Model_Output	Original = 0, and Paraphrased = 1
Loss function	Cross-Entropy
Learning_rate	0.001
Model_optimizer	Adam
Batch size, number of epochs	32, 180
ReLU	Activation function

Table 2. Environment setting and model parameters.

Hyperparameter tuning

In fact, the results of deep learning models are highly dependent on the hyperparameters which are the configuration specifications of the model. In this work, we followed a systematic procedure to choose suitable hyperparameters of the BiLSTM model for Urdu paraphrase detection.

In the process of model parameter tuning, we utilized both the process of turning by hand and turning by an algorithm, including a grid search algorithm as well as a random search algorithm. The key hyperparameters we focused on included:

Learning Rate: Optimization, or, in other words, the Gradient Descent Step determines how big the steps are and when the model should converge.

Batch Size: The notion of batch size refers to the training examples that are processed together in one iteration through the optimization process, in an optimal manner with the right size to minimize computation plus maximize gradients.

Embedding Dimensions: It also reveals how the unit-ness of the word vectors affects the model's capacity to represent semantic and syntactic patterns grounded on the Urdu text.

LSTM Units: The number of units in the LSTM layers determines the model's ability to memorize the long contexts which are very important in paraphrase detection.

Dropout Rates: The dropout rates govern the amount of regularization applied during learning so as to avoid over-fit while learning how to generalize.

We also selected a fixed set of hyperparameters and tested the model on different combinations for tuning the parameters for overall performance, although we used the validation data for generalization accuracy. Thanks to the obtained validation scores, these hyperparameters were tweaked in an iterative manner with the settings applied.

We also applied techniques such as learning rate schedulers and early stopping to further augment the training process. Learning rate schedulers bring the learning rate to a desired schedule that can help the convergence easier and has a better exploration of the parameter space. Recall that, in the previous subsection, we have spoke about early stopping as the practice that helps to avoid overfitting by stopping the model training when for the validation dataset the performance decreases.

We finetuned our BiLSTM framework using different hyperparameters and methods such as learning rate schedulers and early stopping to improve the performance of the models to the best of our ability for the Urdu paraphrase detection task.

Results and discussion

To address this problem, we developed a custom dataset for the Urdu language called the Urdu Paraphrased Corpus (UPC) for this study. This dataset is a huge set of sentence pairs labeled as paraphrased or non paraphrased. For this purpose, data was collected from various written sources available in Urdu including books, blogs, social networking sites, news articles and any written content available on internet. In this study, the data collection process was employed by five native Urdu speaking and paraphrasing individuals. Source texts were given and subjects were told to rewrite the given text while retaining the meaning and using esp. synonym substitution, new sentence construction and summarization approaches. The final corpus comprises of approximately $30{,}087$ sentence pairs; of these, $16{,}715$ sentence pairs have been annotated as paraphrase by human annotators. Besides, with the aim of assessing the efficacy of our suggested BiLSTM model, the dataset adopted in the study is the Quora dataset comprising of sentence pairs from the Question Answering platform where the duplicates are paraphrased sentence pairs or not. This enabled us to determine generalization of the algorithm to real datasets or real-world data.

Limited access to such resources as annotated paraphrasing datasets and pre-trained language models is a considerable challenge to developing reliable paraphrase detection for the Urdu language. Some of the procedures are followed to detect Urdu language paraphrases employing a BiLSTM network. We developed a labelled dataset of Urdu text pairs, each consisting of two sentences: the original and its paraphrase The. The sentences are preprocessed by tokenization, stop word removal and converting the text data into a numeric format using word embeddings. To analyze the model's performance, divide the dataset into 80% for training, 10% for validation, and 10% for testing. For the BiLSTM model, construct input sequences and appropriate labels. Concatenate the two sentences for each pair and get a binary label, 1 for paraphrases and 0 for non-paraphrases. The BiLSTM network generates a binary classification prediction from the concatenated input sequences. The BiLSTM layer enables the model to obtain context in both the forward and backward directions of the input sequence. Train the BiLSTM model with the training dataset and test its performance with the validation dataset.

The results for paraphrase detection using Urdu language text are shown in Table 3. To validate the model performance, we compute the precision, recall, and F-1 score of each baseline model using the custom dataset, shown in Table 4. The CNN model obtained an F-1 score of 85.73 and performed well when processing spatial data, such as pictures, but fail to handle sequential data, such as text, which accounts for their generally poor performance. The LSTM model fared better than the CNN model, with an F-1 score of 87.18. The CNN-LSTM hybrid model, which has a better F-1 score of 90.37, incorporates the advantages of both CNNs and LSTMs. BERT is a pre-trained transformer model that received an F-1 score of 88.66. The proposed model beat all other baseline models with an F-1 score of 95.03.

Incorporating the linguistic rule engine played a crucial role in enhancing the performance of our BiLSTM model, particularly in handling exceptional cases during paraphrase analysis. The rule-based system effectively addressed irregularities and edge scenarios, enabling accurate paraphrase detection despite complex morphological variations, syntactic irregularities, and semantic nuances inherent to the Urdu language.

The CNN-LSTM hybrid model outperformed the other baseline model, achieving an accuracy of 91.48% by combining the strengths of both CNNs and LSTM models, displayed in Table 5. Due to variations in design

Text	Paraphrase
Original Urdu text	اس نے لگن سے مطالعہ کیا اور کلاس میں سب سے زیادہ نمبر حاصل کیے۔
Original English text	He studied diligently and obtained the highest scores in the class.
Paraphrase Urdu text	اپنی محنت اور لگن کی وجہ سے اس نے کلاس میں ٹاپ اسکور حاصل کیا۔
Paraphrase English text	Due to his hard work and dedication, he got the top score in the class.
Original Urdu text	سائنس دان نے مفروضے کو جانچنے کے لیے تجربات کیے۔
Original English text	The scientist conducted experiments to test thehypothesis.
Paraphrase Urdu text	محقق نے مفروضے کی تصدیق کے لیے ٹیسٹ کئے۔
Paraphrase English text	The researcher performed tests to confirm the hypothesis.

Table 3. Paraphrase detection using Urdu language text.

Model	Precision	Recall	F-1 Score (%)
CNN	87.34	84.17	85.73
LSTM	86.71	87.65	87.18
CNN-LSTM	89.43	91.32	90.37
BERT	89.72	87.63	88.66
Proposed BiLSTM	94.35	95.71	95.03

Table 4. The F-1 score achieved using the custom dataset.

Model	Accuracy-quora (%)	Exe-time (s)	Accuracy-custom (%)	Exe-time (s)
CNN	87.03	23.45	83.43	21.04
LSTM	90.19	17.89	88.09	15.31
CNN-LSTM	91.48	19.21	90.67	18.58
BERT	89.72	22.16	87.35	23.42
Proposed BiLSTM	95.34	19.11	94.14	16.87

Table 5. Results of different models on quora and custom-built dataset.

and fine-tuning, the BERT model had a slightly lower accuracy score (89.72%) than the LSTM-CNN model. Figures 3 and 4 show the testing accuracy and loss of the baseline models using the Quora dataset. Similarly, Figs. 5 and 6 show the testing accuracy and loss of the baseline models using the custom dataset.

Comparison with state-of-the-art

Figures Since every language differs, it is essential to identify which algorithms are better suited for collecting documents written in a specific language to detect topics with similar characteristics. The proposed method is compared with state-of-the-art techniques, as shown in Table 6. Using the Quora dataset, and we contrasted the outcomes of the proposed methodology with those of the recent and most similar techniques performed for paraphrase detection. The study in[49] used the CatBoost algorithm for paraphrase detection using the Quora dataset. The study conducted in[50] deployed a combination of LSTM-CNN that achieved an accuracy of 87.50% on the Quora dataset.

Their developed model achieved an accuracy of 75.39% with no preprocessing step. The study in[51] used the T5 (Text-to-Text Transfer) Model to perform paraphrase detection. They achieved an accuracy of 87% when using a preprocessing step. For paraphrase detection using the Quora dataset, Manhattan Long Short-Term Memory (MaLSTM) architecture was used to identify question pairs[24]. MaLSTM is a neural network model used to measure the semantic similarity between text sequences. It achieved an accuracy of 90% when using preprocessing steps. In conclusion, Table 6 compares the performance of several research on the Quora dataset, with the suggested BiLSTM model beating the other models and reaching the maximum accuracy of 95.34%.

The proposed BiLSTM model demonstrated strong performance compared to traditional models like CNN and LSTM, particularly in handling sequential dependencies and achieving higher accuracy. To provide a comprehensive evaluation, we compared the BiLSTM model with transformer-based architectures like BERT. Tables 6 and 7 summarizes the results, including accuracy, F1 score, and computational time for both models on the Quora dataset. While BERT achieved slightly higher accuracy due to its advanced self-attention mechanisms, the BiLSTM model provided competitive performance with significantly lower computational costs. This highlights its practicality for resource-constrained scenarios, such as low-resource language processing tasks. Future work should further benchmark the BiLSTM system against large language models to explore its scalability and robustness across diverse NLP tasks.

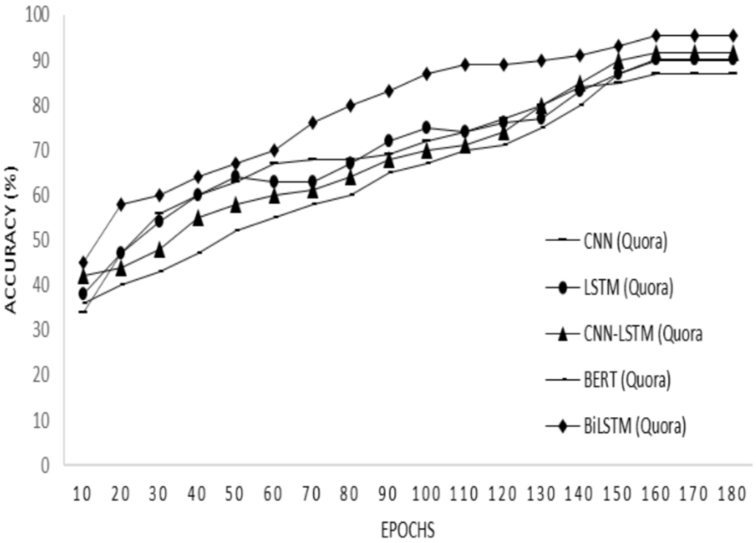

Fig. 3. Testing accuracy obtained using quora dataset.

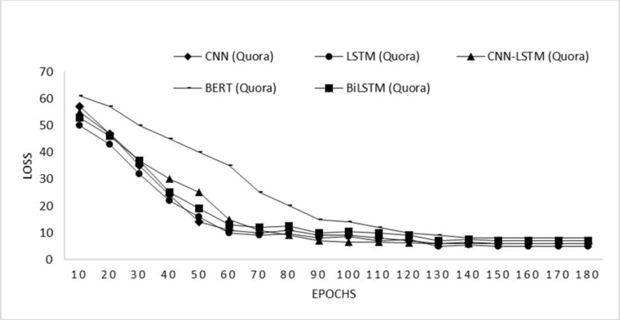

Fig. 4. Testing loss using Quora dataset.

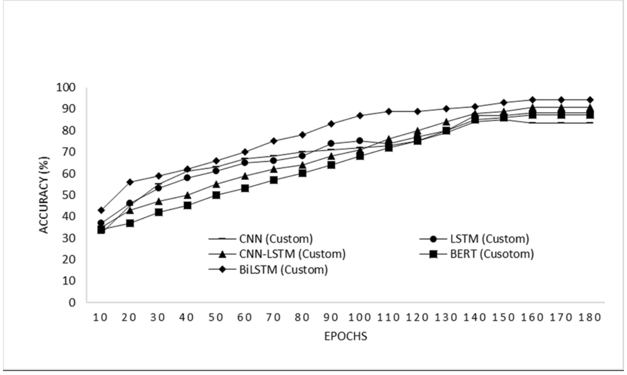

Fig. 5. Testing accuracy achieved using a custom dataset.

Comparison with Urdu paraphrase detection

To validate the model performance, we also compared it with the previously used Urdu paraphrase detection approaches and conducted cross-lingual analysis using Arabic. We used the pre-trained model (BLSTM-API) deployed by Mahmoud et al.[53] for Arabic paraphrase detection. Because this model was trained for the Arabic language, which is similar to Urdu, we obtained the second-highest F-1 score using this model. Table 7 reflects the comparison with the Urdu paraphrase detection task. The transformer model applied in[24] received an F-1 score of 0.63, which is comparatively low. The model's effectiveness suffers if the dataset used to detect Urdu

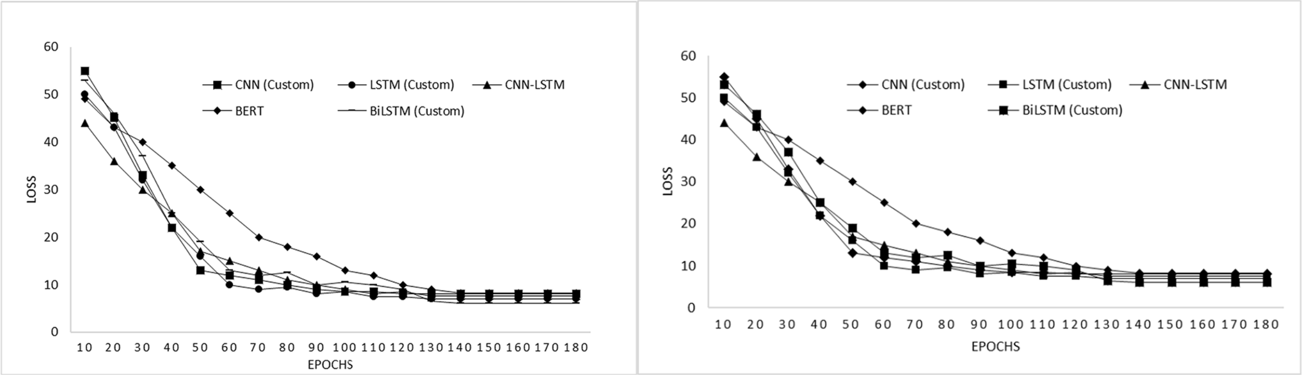

Fig. 6. Testing loss using custom dataset.

Authors	Preprocessing	Approach	Accuracy
Chandra et al.[49]	No	CatBoost	75.39
Mansoor et al.[50]	No	LSTM, CNN	87.50
Palivela et al.[51]	Yes	T5 Model	87.00
Gontumukkala et al.[52]	Yes	MaLSTM	90.00
Proposed	Yes	BiLSTM	95.34

Table 6. Comparison with various related studies for Quora dataset.

Authors	Year	Preprocessing	Approach	F-1 score
Mahmoud et al.[53]	2021	Yes	Pre-trained model	87.32
Mehak et al.[24]	2023	No	Transformer model	63.00
Hafeez et al.[54]	2023	No	Feature fusion	85.00
Proposed	2023	Yes	BiLSTM	95.03

Table 7. Comparison with works done for Urdu paraphrase detection.

paraphrases is small, imbalanced, or noisy. The feature fusion technique employed in[54] produced an F-1 score of 0.85. The suggested BiLSTM model possessed the highest F-1 score of 95.03. Because BiLSTM can successfully capture long-range relationships in text, it is well-suited for sequence-to-sequence problems like paraphrase identification.

Limitations

When trained on a general-purpose text like English, the suggested model may perform poorly in specialized domains, such as Urdu or Arabic. Further data and fine-tuning are needed to adjust the model to domain-specific applications. During the experiments, we found that changing the BiLSTM model to shifting language patterns or novel words is complex and may need regular retraining. The proposed method is suitable for other languages, such as Arabic.

Error analysis

An error analysis into the evaluation reveals specific BiLSTM model difficulties although its accuracy is high. Some semantic similarity misclassification errors were also noted which affected pairs of similar meaning but different wordings due to limited context. Phrasemes, which can convey their meanings only in certain contexts as well as syntactically complex constructions containing many syntactic dependencies or deep syntactic structures were difficult. These problems are primarily due to two major aspects: firstly regarding the contextual coverage of BiLSTM, and secondly due to the complex morphological structure of Urdu. Some of these future works can help to overcome above challenges include the addition of attention mechanism that will address long range dependencies of the sequence and adopting bigger linguistic rule engine to get better at idiomaticity.

Conclusions and future directions

Paraphrase detection is a crucial NLP task in plagiarism detection, text content summarization, information retrieval, and question-answering systems. Detection becomes problematic in the case of Urdu language text because of a lack of resources and the complex morphological structure of Urdu. In this study, we successfully created a BiLSTM framework-based paraphrase detection system customized to the Urdu language text. We fine-tuned the BiLSTM model, allowing us to efficiently capture contextual dependencies, achieve promising results, and outperform state-of-the-art works. We also observed that the performance of the BiLSTM framework heavily relies on the size and diversity of the training data. Therefore, we created a more diverse dataset of Urdu paraphrase pairs. We compared the model performance with the baseline approaches and Urdu paraphrase detection tasks to validate it.

While the proposed BiLSTM-based framework has shown promising results for Urdu paraphrase detection, it is essential to acknowledge and address the limitations mentioned in this study to improve the proposed idea further. To this end, we outline the following potential enhancements: (1) Incorporate Transfer Learning by pre-training the BiLSTM model on an extensive general Urdu corpus and then fine-tuning it on the paraphrase detection task, which could enhance generalization and adaptation to diverse domains, addressing the limitation of domain-specific performance; (2) Integrate Attention Mechanisms into the BiLSTM architecture, enabling the model to better focus on the most relevant parts of the input text, potentially improving the capture of long-range dependencies and contextual information, thereby addressing the limitation of handling structural variations in input sentences; (3) Develop Domain-Adaptive Models by fine-tuning the BiLSTM architecture on domain-specific corpora, such as academic texts, news articles, or social media content, which could enable better adaptation to the linguistic nuances and terminology of each domain, improving paraphrase detection accuracy within those contexts. By incorporating these enhancements, our approach can effectively address the limitations mentioned in this study, refining the proposed idea and advancing the state-of-the-art Urdu paraphrase detection while contributing to the broader development of NLP capabilities for this language. The proposed BiLSTM framework has been validated for its contributions to Urdu paraphraset level detection in context of sentence level Urdu, while the ensuing research developments can broaden the added tasks and languages. One such direction is the generalization to the more difficult paradigm of paragraph-level paraphrase detection which poses its own problems such as keeping the coherence of the transformed context at the paragraph level and understanding the more difficult level of similarity. There are some possibilities that might help to overcome these challenges, to look further at the development of more complex architectures, for example, at hierarchical models or transformers. Furthermore, the presented BiLSTM framework and linguistic rule engine can be tested in other low-resource languages such as Pashto, Sindhi, or Amharic in order to expand the applicability of this work to other diverse linguistic tasks. The above sort of approaches would go a long way in enhancing the progressive work on natural language processing in relatively minor languages.

In the future, in order to improve the model, the authors of the work would like to add attention mechanisms to the BiLSTM architecture to help the model decide in which parts of the input text it is most important to detect paraphrases in. It will allow having various weights for various phrases or tokens in the input and may increase the overall performance and reduce the losses of long-range dependencies. Additional improvements include incorporating the attention mechanisms or a hybrid architecture as a next step to improving the overall performance and complexity of the model. BiLSTM combined with transformers or any other architecture seems to be the right direction to balance the complexity of language phenomena and computations at the same time. These technical directions, there is a need to perform proofread this paper well to fix other problems of typographical errors and structural conflicts. Adhering to consistent formats partly in the tables and figures, headings and references section will enhance clarity and professionalism of the work an idea which will supplement its common overall comparative advantage in the academic field a question of replicability in the academic field.

Data availability

The data may be provided upon reasonable request to corresponding author.

Received: 3 July 2024; Accepted: 5 March 2025
Published online: 02 May 2025

References

1. Dinh, D. & Thanh, N. L. English–Vietnamese cross-language paraphrase identification using hybrid feature classes. *Heuristics* **28**, 193–209 (2022).
2. Altheneyan, A. & Menai, M. E. B. Evaluation of state-of-the-art paraphrase identification and its application to automatic plagiarism detection. *Pattern Recognit. Artif. Intell.* **34**(40), 2053004 (2020).
3. Yang, Y., Zhang, Y., Tar, C. & Baldridge, J. A cross-lingual adversarial dataset for paraphrase identification. *arXiv preprint arXiv:1908.11828* (2019).
4. Chingacham, A., Demberg, V. & Klakow, D. Exploring the potential of lexical paraphrases for mitigating noise-induced comprehension errors. *arxiv Preprint arxiv:2107.08337* (2021).
5. Shahinmoghadam, M., Ebrahimi Kahou, S. & Motamedi, A. Neural semantic tagging for natural language-based search in building information models: Implications for practice. *Comput. Ind.* **155**, 104063 (2024).
6. Iqbal, H. R., Qadir, M. U., Aslam, S. & Usman, T. Urdu paraphrase detection: A novel DNN-based implementation using a semi-automatically generated corpus. *Nat. Lang. Eng.* **30**(2), 354–384 (2024).
7. Taha, K., Hassan, F., Singh, J. & Wang, M. Y. Text classification: A review, empirical, and experimental evaluation. *arXiv preprint arXiv:2401.12982*. (2024).
8. Iqbal, H. R., Maqsood, R., Raza, A. A. & Hassan, S. Urdu paraphrase detection: A novel DNN-based implementation using a semi-automatically generated corpus. *Nat. Lang. Eng.* 1–31 (2023).
9. Radford, A. et al. Language models are unsupervised multitask learners. *Open AI Blog* **1**, 9 (2019).
10. Sanh, V., Debut, L., Chaumond, J. & Wolf, T. Distil BERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*. (2019).
11. Medvecki, D., Mladenovic, N. & Mitrovic, I. Multilingual transformer and BERTopic for short text topic modeling: The case of Serbian. In *Conference on Information Technology and its Applications* (Springer, Cham, 2024).
12. Wang, X. & Wang, B. Exploring automatic methods for the construction of multimodal interpreting corpora. How to transcribe linguistic information and identify paralinguistic properties? *Lang. Cultures* (2024).

13. Putra, I. M. S., Siahaan, D. & Saikhu, A. SNLI Indo: A recognizing textual entailment dataset in Indonesian derived from the Stanford Natural Language Inference dataset, *Data Brief.* **52**, 109998, (2024).
14. Muneer, I., Ghani, H. U., Shoaib, S. & Rajput, K. B. Developing a large benchmark corpus for Urdu semantic word similarity. *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, **22**(3), 1–19 (2023).
15. Jain, R., Gupta, S. & Kumar, R. ParaCap: paraphrase detection model using capsule network. *Multimed. Syst.* 1–19, (2022).
16. Deng, J., Wu, Z., Sun, X. & Ma, Y. Milmo: minority multilingual pre-trained language model. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (IEEE, 2023).
17. Hunt, E. et al. l, Machine learning models for paraphrase identification and its applications on plagiarism detection. In *Proceedings of the 2019 IEEE International Conference on Big Knowledge (ICBK)* 97–104 (Beijing, China, 2019).
18. Shahmohammadi, H., Dezfoulian, M. & Mansoorizadeh, M. Paraphrase detection using LSTM networks and handcrafted features. *Multimed. Tools Appl.* **80**, 6479–6492 (2021).
19. Gangadharan, V., Gupta, D., Amritha, L. & Athira, T. Paraphrase detection using deep neural network-based word embedding techniques, In *Proceedings of the 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)* 517–521 (Tirunelveli, India, 2020).
20. Vrbanec, T. & Meštrović, A. Corpus-based paraphrase detection experiments and review. *MDPI* **11**(5), 241 (2020).
21. Munaf, M., Shahid, N., Sabir, S. Z., Asad, M. Z. & Kamal, K. M. Low resource summarization using pre-trained language models. *arXiv preprint arXiv:2310.02790*. (2023).
22. Desouki, M. I. E., Gomaa, W. H. & Abdalhakim, H. A hybrid model for paraphrase detection combines pros of text similarity with deep learning. *Int. J. Comput. Appl.* **178**(20), 18–23 (2019).
23. Ahmed, M., Samee, M. R. & Mercer, R. E. Improving tree-LSTM with tree attention. In *Proceedings of the 2019 IEEE 13th International Conference on Semantic Computing (ICSC)* 247–254 (Newport Beach, CA, USA, 2019).
24. Mehak, G., Muneer, I. & Nawab, R. M. A. Urdu text reuse detection at phrasal level using sentence transformer-based approach. *Expert Syst. Appl.* **234**, 121063 (2023).
25. Tang, R., Chuang, Y. N. & Hu, X. The science of detecting LLM-generated text. *Commun. ACM* **67**(4), 50–59 (2024).
26. Bölücü, N., Can, B. & Artuner, H. A Siamese neural network for learning semantically-informed sentence embeddings. *Expert Syst. Appl.* **214**, 119103 (2023).
27. Shajalal, M. & Aono, M. Semantic textual similarity in bengali text. In *Proceedings of the 2018 International Conference on Bangla Speech and Language Processing (ICBSLP)* 1–5 (2018).
28. Zuo, F. et al. Neural machine translation inspired binary code similarity comparison beyond function pairs. *arXiv preprint arXiv: 1808.04706* (2018).
29. de Dios Zapata, E., Cornejo, P., García-Marirrodriga, J., Galdón-Quiroga & Reynolds-Barredo, J. M. A novel unsupervised machine learning algorithm for automatic Alfvénic activity detection in the TJ-II stellarator (2024).
30. Altalib, M. K. & Salim, N. Similarity-based virtual screen using enhanced siamese deep learning methods. *ACS Omega* **7**(6), 4769–4786 (2022).
31. Bao, W., Bao, W., Du, J., Yang, Y. & Zhao, X. Attentive Siamese LSTM network for semantic textual similarity measure. In *Proceedings of the 2018 International Conference on Asian Language Processing (IALP)* 312–317 (Bandung, Indonesia, 2018).
32. Devlin, J., Chang, M. W., Lee, K. & Toutanova, K. Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:04805*. (2018)
33. Roman, M. Citation intent classification using word embedding. *IEEE Access* **9**, 9982–9995 (2021).
34. Elhassan, N. et al. Arabic sentiment analysis based on word embeddings and deep learning. *Computers* **12**, 126. https://doi.org/10.3390/computers12060126 (2023).
35. El-Affendi, M. A., Khawla, A. & Amir, H. A novel deep learning-based multilevel parallel attention neural (MPAN) model for multidomain Arabic sentiment analysis. *IEEE Access* **9**, 7508–7518 (2021).
36. Ghani, M. A. N. U. et al. Toward robust and privacy-enhanced facial recognition: A decentralized blockchain-based approach with GANs and deep learning. *Math. Biosci. Eng.* **21**(3), 4165–4186. https://doi.org/10.3934/mbe.2024184 (2024).
37. Ghani, M. A. N. U. et al. Enhancing security and privacy in distributed face recognition systems through blockchain and GAN technologies. *Comput. Mater. Contin.* **79**(2), 2609–2623. https://doi.org/10.32604/cmc.2024.049611 (2024).
38. Ghani, M. A. N. U. et al. Securing synthetic faces: A GAN-blockchain approach to privacy-enhanced facial recognition. *J. King Saud Univ.-Comput. Inf. Sci.* **36**(4), 102036. https://doi.org/10.1016/j.jksuci.2024.102036 (2024).
39. Wang, H., Li, Y., Zhao, K. & Chen, J. DAFA-BiLSTM: Deep autoregression feature augmented bidirectional LSTM network for time series prediction, neural networks, **157**, 240–256 (2023).
40. Farooq, E. & Borghesi, A. A federated learning approach for anomaly detection in high performance computing. In *2023 IEEE 35th International Conference on Tools with Artificial Intelligence (ICTAI)* (IEEE, 2023).
41. Zhou, C., Qiu, C. & Acuna, D. E. Paraphrase identification with deep learning: A review of datasets and methods. *arXiv preprint arXiv: 2212.06933* (2022).
42. Kang, N., Singh, B., Afzal, Z., van Mulligen, E. M. & Kors, J. A. Using rule-based natural Language processing to improve disease normalization in biomedical text. *J. Am. Med. Inform. Assoc.* **20**(5), 876–881 (2013).
43. Ledig, C. et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of CVPR* 4 (2017).
44. Khan, S. et al. A novel transformer attention-based approach for sarcasm detection. *Exp. Syst.* **13686** (2024).
45. Khan, S. et al. An automated approach to identify sarcasm in low-resource language. *PLoS One* **19**(12), e0307186 (2024).
46. Sharma, L., Graesser, L., Nangia, N. & Evci, U. Natural language understanding with the quora question pairs dataset. *arXiv preprint arXiv: 1907.01041* (2019).
47. Wang, S., Zhou, W. & Jiang, C. A survey of word embeddings based on deep learning. *Computing* **102**, 717–740 (2020).
48. Rahman, A. U. et al. Extended ICA and M-CSP with BiLSTM towards improved classification of EEG signals. *Soft Comput. Vol.* **26**, 10687–10698 (2022).
49. Chandra, A. & Stefanus, R. Experiments on paraphrase identification using quora question pairs dataset, arxiv Preprint arxiv: Cs. (2020). CL/2006.02648.
50. Mansoor, M., Ur Rehman, Z., Shaheen, M., Khan, M. A. & Habib, M. Deep learning based semantic similarity detection using text data. *Inform. Technol. Control.* **49**(4), 495–510 (2020).
51. Palivela, H. Optimization of paraphrase generation and identification using language models in natural language processing. *Int. J. Inform. Manag. Data Insights* **1**(2), 100025 (2021).
52. Gontumukkala, S. S. T. et al. Quora question pairs identification and insincere questions classification. In *Proceedings of the 13th International Conference on Computing Communication and (ICCCNT)* 1–6 (Kharagpur, India, 2022).
53. Mahmoud, A. & Zrigui, M. Bi-LSTM recurrent neural network-based approach for Arabic paraphrase identification. *Arab. J. Sci. Eng.* **46**, 4163–4174 (2021).
54. Hafeez, H., Muneer, I., Sharjeel, M. & Ashraf, M. A. R. M. Adeel Nawab, Urdu short paraphrase detection at sentence level. *ACM Trans. Asian Low-Resource Lang. Inform. Process.* **22**(4), 1–20 (2023).

Acknowledgements

This study is supported via funding from Prince sattam bin Abdulaziz University project number

(PSAU/2024/R/1445).

Author contributions

M.A.A.: Being a first author of the paper, I have proposed and implement the conceptualization of the work. Moreover, He wrote the first draft of the paper as well. K.U.K.: Data collection and interpretation. W.K.: analysis and curation of the data done by S.U.K. The corresponding author and analyze the overall structure of the paper. S.U.K.: Methodology, software, validation, and analytical review of the paper. A.A.: Formal analysis, Data validation, Project supervision are done by him. S.A.A.: Performed the major tasks in revised version. Adding details to Corpus creating process. worked on Error analysis part, and the overall revised version was administered and proof read by him.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to S.U.K.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.