Complex & Intelligent Systems (2025) 11:489
https://doi.org/10.1007/s40747-025-02066-6

ORIGINAL ARTICLE

Advancing Urdu named entity recognition: deep learning for aspect targeting

Kamran Aziz[1] · Naveed Ahmed[2] · Yaoxiang Yu[3] · Hassan Jalil Hadi[2] · Mohammaed Ali Alshara[2] · Umair Tariq[4,5] · Donghong Ji[3]

Received: 10 December 2024 / Accepted: 18 August 2025 / Published online: 29 October 2025

Abstract

This study unveils the Named Entity Recognition (NER) system specifically designed for Urdu news headlines, aimed at bridging crucial linguistic resource gaps. We meticulously developed a comprehensive corpus from diverse news sources, specifically tailored to reflect Urdu's unique orthographic and morphological characteristics. Our approach incorporates state-of-the-art (SOTA) neural technologies including transformers for deep contextual embeddings, Graph Convolutional Networks (GCN) for detailed syntactic analysis, and Biaffine Attention mechanisms to enhance inter-token relationships. A Conditional Random Field (CRF) layer further ensures accurate and consistent entity labeling, improving the system's precision. Initially, our model was rigorously benchmarked using established transformer models such as XLM-R, mBERT, and XLNet to set initial performance benchmarks. Subsequent enhancements involved integrating encoder functionalities from generative models like mBART and mT5, allowing a thorough comparative evaluation of these advanced encoders against our benchmarks. This phase aimed to assess their potential in effectively detecting implicit entities, thus enhancing our model's functionality for complex searches and automated content categorization on Urdu digital platforms. Our improvements notably contribute to computational linguistics by extending SOTA language technologies to under-resourced languages and promoting greater inclusivity in Natural Language Processing (NLP).

Keywords Named Entity Recognition · Data mining · NLP · Entity extraction · XLM-R · Deep learning

Introduction

NER is a critical area of research in the field of NLP, focusing on the detection and categorization of semantic labels from text into predefined groups [1]. These groups typically encompass proper names, such as persons, organizations, and geographical locations, as well as other noteworthy identifiers like dates, financial figures, and technical terms. NER serves as a foundation for a myriad of advanced NLP applications, including but not restricted to semantic search, content analysis, and knowledge graph construction [2, 3]. By transforming raw text into a form that machines can analyze with a degree of human-like understanding, NER encourages more intelligent and context-aware processing of large volumes of natural language data, which is invaluable in the age of information overload [4–6]. Figure 1 provides an overview of the NER process, showcasing how entities are identified and categorized within a piece of text.

Building on the importance of NER in various NLP tasks, a notable area of research is dedicated to enhancing

✉ Donghong Ji
dhji@whu.edu.cn

1 Digital Technologies, Hainan Bielefeld University of Applied Sciences, Hainan 578001, China

2 College of Computer and Information Sciences, Prince Sultan University, Riyadh, Saudi Arabia

3 Key Laboratory of Aerospace Information Security and Trusted Computing Ministry of Education, School of Cyber Science and Engineering, Wuhan University, Wuhan, China

4 College of Computing, Faculty of Computing Engineering and the Built Environment. Birmingham City University, Birmingham, UK

5 Department of Digital Media Technology, Birmingham Institute of Fashion and Creative Arts (BIFCA), Wuhan Textile University, Wuhan, China

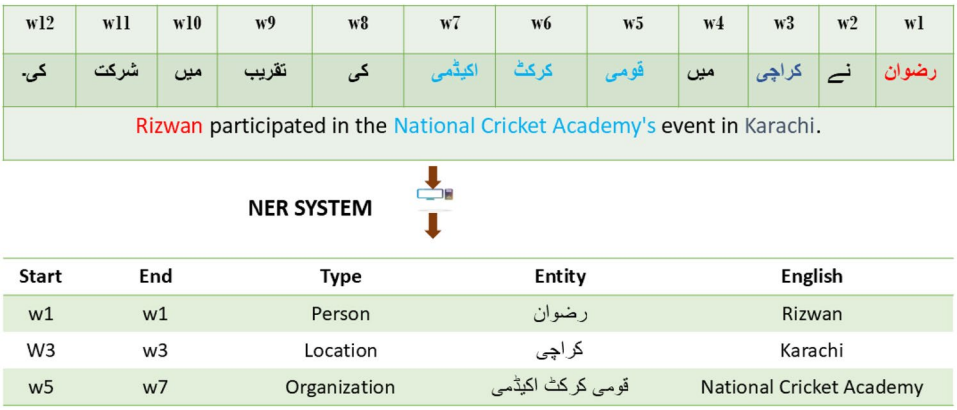

Fig. 1 An illustration of NER task for a News Headline

the accuracy and adaptability of NER systems across different languages and domains [3, 7, 8]. This involves refining algorithms and models to better capture context and disambiguate meanings, as well as creating comprehensive and annotated corpora for valuable training. For languages like Urdu, with its rich morphological structures and complex syntax, developing strong NER systems requires an extensive understanding of linguistic features and a customized approach to model architecture [9]. As computational capabilities and machine learning techniques advance, the field is dynamically evolving to create more nuanced NER systems capable of navigating the intricacies of human language [2, 10].

NER faces unique challenges due to the language's complex script, morphological complexity, and syntactic ambiguity. The Urdu script, influenced by Persian and Arabic, features a connected cursive form and complex orthography that, combined with the language's flexible grammar and variable word order, entangles the task of accurately identifying entities. [11–13]. The deficiency of large, annotated datasets for Urdu amplifies the problem, as machine learning models thrive on abundant data for training. As a result, efforts in Urdu NER are conducted towards developing bespoke algorithms that can navigate these linguistic intricacies and datasets that capture the broad stylistic and dialectal diversity of the language, making the accurate recognition of entities possible [14–17].

The development in NER systems marks a transition towards greater linguistic precision and understanding [18–21]. This is particularly vital for Urdu, a language steeped in poetic tradition and characterized by a variety of linguistic registers. Current NER systems are being designed to grasp the full spectrum of Urdu's linguistic diversity, including its formal and colloquial nuances, honorifics, and the intricacies of its script [22, 23]. Leveraging contextual embeddings and deep learning frameworks, researchers are overcoming the challenges posed by script variations and orthographic complexity. Integrating language-specific resources with cutting-edge NLP models is paving the way for advanced applications in information extraction for Urdu, enhancing search engines, text analytics, and AI-driven linguistic research [3, 7, 24, 25].

Our NER model utilizes SOTA neural network techniques to adeptly handle the linguistic complexities inherent in the Urdu language. We employ mBERT to provide deep contextual embeddings, capturing a broad spectrum of linguistic nuances, including those crucial for understanding cross-lingual contexts relevant to Urdu. To further enhance our model's ability to recognize entities, we exploit syntactic structures using GCN, and we incorporate Biaffine Attention mechanisms to precisely discern inter-token relationships. Biaffine Attention is particularly effective in our context for modeling pairwise interactions between tokens, which is essential for handling the complex syntactic constructs typical of Urdu. Additionally, a CRF layer ensures precise and coherent labeling of entities, enhancing the system's reliability. This vigorous system sets a new benchmark in Urdu Named Entity Recognition (UNER), contributing significantly to computational linguistics and improving content categorization and search functionality within Urdu digital media. It advances the SOTA in NER for low resource languages with its futuristic application of these technologies:

- We developed a linguistically rich corpus by collecting Urdu news headlines from diverse sources, ensuring comprehensive coverage of orthographic, morphological, and syntactic variations. This corpus provides a strong foundation for effective NER by addressing the unique challenges posed by the Urdu language.
- Our model leverages deep contextual embeddings from transformer-based models like BERT, along with syntactic information modeled via GCN. To further enhance token-level interaction modeling, we integrate a Biaffine Attention mechanism and a CRF layer, which collectively improve accuracy and ensure coherent entity labeling.

- To explore performance on more semantically implicit structures, we also assess the utility of generative models such as mBART and mT5. These models offer promising capabilities for recognizing complex and contextually embedded named entities, thereby broadening the applicability of our system across diverse linguistic contexts.

Literature review

For the past several decades, extensive research and application of NER systems for English have taken place. This has led to the development of various NER approaches, including machine learning, rule-based, and hybrid systems [3, 7, 26]. However, due to the distinct characteristics of Urdu, NER systems designed for English and other European languages are not directly transferable. Despite this, progress has been made in Urdu NER, primarily through rule-based and machine learning methodologies [27, 28]. Table 9 presents a comparative analysis of various studies conducted on NER in the Urdu language.

Rule-based approaches

Riaz et al. developed a rule-based NER system that classifies six types of named entities and tested it on 2262 documents from the Becker–Riaz corpus [29, 30]. Their system outperformed statistical models, achieving high precision, recall, and F-scores of 90.7%, 91.5%, and 90.7%, respectively. However, the system's performance is limited to the specific corpus and does not generalize well to other texts. It also lacks capabilities to detect rare words and nested named entities due to its reliance on a custom gazetteer.

Singh et al. developed a rule-based NER system for Urdu that identifies thirteen named entity types using dictionaries, lexicons, and affix lists [31]. The system relies on domain-specific gazetteers and syntactic-lexical patterns [32, 33]. Evaluated on two datasets from BBC Urdu, the system achieved varying levels of precision, recall, and F-scores, with limitations including a fixed window size and inconsistent handling of multi-word date formats (see Table 1).

Table 1 Comparative analysis of Urdu NER approaches

Study	Approach	Strengths	Limitations
Riaz et al. [29, 30]	Rule-based	High precision and recall on specific corpus	Poor generalizability; no support for rare/nested entities
Singh et al. [31]	Rule-based	Utilizes gazetteers and affix lists; identifies 13 NE types	Limited to fixed window size; issues with multi-word patterns
Jahangir et al. [34]	Statistical (n-gram)	Demonstrates potential for statistical methods in Urdu	Performance may vary across datasets; needs gazetteer support
Malik and Sarwar [35]	HMM	Includes POS tagging and rule-based enhancements	Sensitive to tagging scheme; moderate complexity
Riaz et al. [36]	Maximum Entropy	Tackles structural challenges of Urdu	Limited by data sparsity and lack of resources
Khan et al. [37]	CRF	Uses UNER-I dataset; rich feature engineering	Requires annotated corpus; computationally intensive
Haq et al. [12]	Deep Learning (BiLSTM)	Automated feature learning; improved F1-score	Requires large datasets; lacks interpretability
Anam et al. [38]	Deep Learning (BiLSTM+GRU with Floret/FastText embeddings)	Strong performance with contextual embeddings; robust across datasets	Dependent on pre-trained embeddings; generalization to noisy text uncertain
Ullah et al. [39]	Deep Learning with Data Augmentation	Introduces CWEA for enriching Urdu datasets; enhances transformer model performance	Effectiveness may vary across domains; augmentation quality critical
Ullah et al. [40]	Attention-BiLSTM-CRF	Embedding-level attention; high performance on benchmark corpus	Requires high compute; risk of overfitting small datasets
Biswas et al. [41]	Hybrid	Combines rule-based and statistical learning	Complex integration and tuning needed
Saha et al. [42]	Hybrid	Language adaptability; handles multilingual data	Requires domain-specific rules; resource heavy
Gali et al. [43]	Hybrid	Uses transliteration and multiple features for NE extraction	Depends on high-quality linguistic resources

Machine learning approaches

Jahangir et al. developed a statistical NER system using unigram and bigram n-gram models enhanced with gazetteer lists and smoothing techniques. Their system demonstrated the potential of statistical methods in NER for under-resourced languages like Urdu [34]. Malik and Sarwar developed a system utilizing a Hidden Markov Model (HMM) targeting entities such as persons, organizations, and locations. They compared the IOB2 and IOE2 tagging schemes and found the latter to be more effective in their preprocessing steps for Urdu [35]. Their system also incorporates Part of Speech tagging, gazetteers, and rule-based enhancements to improve accuracy.

Riaz et al. explored Urdu NER using a maximum entropy model, addressing the challenges of NER for a language with insubstantial resources and complex structure. Their experiments on the IJCNLP 08 dataset highlighted the developmental stage of NER technologies for South Asian languages, particularly Urdu [36].

Khan et al. developed a CRF-based approach incorporating both language-specific and language-independent features, and created the UNER-I dataset for Urdu NER. Their method outperformed baseline models, enhancing F1 scores noteworthy [37].

Deep learning approaches

Haq et al. developed deep learning models for Urdu NER that automate feature extraction, minimizing reliance on manual engineering. Their approach, tested on multiple datasets, demonstrated an important improvement in F1 score [12].

Ullah et al. introduced an advanced Attention-Bi-LSTM-CRF model applied to the MK-PUCIT Corpus, integrating word-level embeddings with a novel embedding-level focus mechanism, achieving a notable F1-score of 92% [40].

Ullah et al. (2024) introduced a novel data augmentation technique, Contextual Word Embeddings Augmentation (CWEA), to improve NER for low-resource languages like Urdu [39]. By enriching the training data, they demonstrated how transformer-based models such as BERT and RoBERTa can benefit from augmented datasets. Their work emphasizes the importance of data expansion and contextual representation for achieving more accurate and robust NER in complex linguistic settings.

Anam et al. (2024) proposed a deep learning-based approach for Urdu NER that leverages FastText and Floret word embeddings to enhance contextual feature representation. These embeddings are fed into various neural architectures, including LSTM, BiLSTM, GRU, and CRF models, to perform sequence labeling tasks. Their method demonstrates strong performance and robustness across multiple benchmark Urdu datasets, emphasizing the effectiveness of contextual word embeddings and recurrent neural networks in handling the linguistic challenges of low-resource languages like Urdu [38].

Hybrid approaches

Hybrid NER techniques integrate rule-based and statistical methods. Notable efforts include the combination of maximum entropy and Hidden Markov Models, and the development of multilingual NER systems incorporating transliteration for languages including Hindi and Urdu [41–46]. These strategies demonstrate the adaptability and effectiveness of hybrid NER in handling complex linguistic features.

Properties impacting named entity recognition

NER in Urdu faces unique challenges due to the intrinsic properties of the language and its script. These properties influence the identification and classification of named entities.

Limited annotated corpora

Urdu is considered a low-resource language in terms of annotated datasets for NER tasks. This lack of large, high-quality corpora poses significant challenges for training supervised models effectively [47].

Ambiguous word boundaries

Word boundary detection in Urdu is hindered by the inconsistent use of spaces and the agglutinative nature of the language. Named entities often appear embedded within compound words, making entity segmentation more complex [48].

No capitalization

Urdu does not utilize capitalization, which is common in many South Asian languages. This characteristic makes it challenging to automatically recognize proper nouns, which are often capitalized in Western languages. For instance, while the acronym 'UNESCO' is always capitalized in English, its Urdu equivalent, یونیسکو, appears without any capitalization, posing difficulties for automated recognition systems [49].

Morphological complexity

The extensive use of affixes in Urdu adds to its morphological complexity. This can obscure the boundaries of named

entities in texts. For example, the name کراچی (Karachi) may be expanded with the locative suffix میں (in) to form کراچی میں, making it harder to separate the proper name from grammatical affixes [39].

Spelling variations

The absence of standardized spelling in Urdu leads to multiple valid representations of the same word, which complicates text processing. For example, the word 'Lahore' can be spelled as لاہور, لہور, or لاہر, and each variation needs to be recognized by the NER system as the same location [13, 48].

Lexical borrowings

Urdu's lexicon includes a great number of loanwords from Persian, Arabic, Turkish, and English, which introduces additional challenges for NER systems due to diverse phonetic and morphological patterns. For example, the English word 'bank' is used in Urdu as بینک, and the Arabic origin word کتاب (book) varies from native Urdu words [12, 49].

Free word order

The flexibility in Urdu sentence structure allows variation in word order, complicating the syntactic parsing necessary for effective entity recognition. For example, the sentence آمنہ اسکول گئی (Amna went to school) can also be written as اسکول آمنہ گئی, where the position of the subject and the location are reversed, impacting the performance of entity recognition algorithms [47, 50]. These features of the Urdu language illustrate the need for specially designed NER systems that consider these complexities to achieve effective recognition of entities.

Table 2 Statistics of the collected dataset

Domain	Online source
Politics	Ary.com, urdu.news18.com, voa.com, bbc.com/, humnews.pk
International Affairs	Ary.com, Ajjnews.com, humnews.pk
Sports	geo.tv, Ary.com
Entertainment	Ary.com, humnews.pk
Technology	humnews.pk

Corpus acquisition

The dataset utilized for NER was meticulously curated from several prominent Pakistani news sources, including Hum News, BBC Urdu, ARY News, VOA Urdu News, GEO News, and Aaj News. The data collection process prioritized the inclusion of four specific categories: political news, sports news, entertainment, and technology news. This diverse selection ensures the representation of a broad spectrum of sentiments across different subjects, with political news often exhibiting strong sentiment polarities, from harsh criticisms of opponents to fervent support for political figures. The annotation of the dataset was carried out by three graduate students who are native Urdu speakers. To ensure the accuracy and consistency of the annotations, a graduate student specialized in Urdu language further reviewed and validated the annotations in detail. Table 3 presented few examples for annotation being done on news headlines.

Table 2 presents the statistics of the collected dataset, categorizing the data according to domains and the online sources utilized.

Comprehensive annotation guidelines for Urdu news headlines NER

Influence from key studies

- **CoNLL-2003 Shared Task:** Adapted guidelines for annotating standard entity types including persons, locations, and organizations to fit Urdu news [51].
- **ACE 2005 Multilingual Training Corpus:** Utilized to refine entity definitions and ensure comprehensive entity coverage [52, 53].
- **MUC-7:** Provided historical context to NER and influenced decisions on entity boundaries and categorization [54, 55].

Table 3 Detailed examples of Urdu named entities extraction by type

Entity type	Headline	Named entities
Person	وزیراعظم عمران خان نے اجلاس کی صدارت کی۔	وزیراعظم عمران خان
	Prime Minister Imran Khan chaired the meeting	Prime Minister Imran Khan
Location	لاہور میں ثقافتی میلہ منعقد کیا گیا۔	لاہور
	A cultural festival was held in Lahore	Lahore
Organization	یونیسیف نے فنڈ ریزنگ مہم شروع کی۔	یونیسیف
	UNICEF started a fundraising campaign	UNICEF
Time	رمضان کا آغاز بدھ سے ہوگا۔	رمضان
	Ramadan will begin on Wednesday	Ramadan

Entity types and annotation examples

Person: Includes names, titles as direct substitutes for names, e.g., "وزیراعظم" (Prime Minister). Example: "عمران خان کراچی میں" translates to "Imran Khan in Karachi."

Location: Geographical entities like cities and countries. Example: "لاہور میں بارش" translates to "Rain in Lahore."

Organization: Formal groups such as companies and government bodies. Example: "یونیسف نے اعلان کیا" translates to "UNICEF announced."

Time: Time expressions like dates and days. Example: "پیر کو میٹنگ ہوگی" translates to "The meeting will be on Monday."

Miscellaneous: Notable entities not covered by other categories. Example: "موبائل فون کی فروخت" translates to "Mobile phone sales."

Specific guidelines for annotation

- **Designations as Persons:** Titles referring to individuals in news headlines are tagged as persons because in headlines domain often persons and addressed with their titles.
- **Handling of 'Others':** Generic nouns, adjectives, or verbs not fitting other categories are tagged as 'Others.'
- **Consistency and Context:** Annotators ensure consistent application of guidelines, using context for ambiguous terms.
- **Quality Assurance:** The process includes multiple review stages to ensure accuracy, with discrepancies resolved by a senior annotator.

Table 4 Statistics of the NER dataset for Urdu news headlines

Statistic	Value
Total headlines	16,604
Average tokens per headline	13.63
NER Tag Distribution:	
Organization	4000
Location	4235
Person	10,249
Time	661
Total unique NER tags	4

- **Training and Execution:** Annotators receive extensive training on the guidelines and practice on actual data.

Dataset Profile

See Table 4.

To ensure the reliability and constancy of our NER dataset, we used the Cohen's Kappa statistic, a well-known metric for measuring inter annotator agreement. This measure was essential for evaluating how consistently two annotators could classify and label entities within the predefined NER categories: Organization, Location, Person, and Time.

Our NER corpus achieved an overall Inter-Annotator Agreement Cohen's Kappa score of 73.29%, indicating substantial agreement among the annotators [56–58]. This high level of agreement highlights the quality of the annotations and the clarity of the provided annotation guidelines. Such rigorous validation ensures that the dataset is both robust and reliable for training advanced NER models that can perform accurately across diverse Urdu news contexts.

Dataset Details: The Table 2 illustrate 16,604 headlines with an average of 13.63 tokens per headline. It includes:

- **Organization:** 4,000 entries
- **Location:** 4,235 entries
- **Person:** 10,249 entries, reflecting frequent references to individuals in news headlines
- **Time:** 661 entries, marking specific dates and times

This detailed approach to dataset curation not only supports advanced NER research but also contributes to computational linguistics within the Urdu language context.

Proposed methodology

In this section, we will elaborate on the model construction and its working Fig. 5 is the overall structure of this model. Whereas Fig. 4 shown the preprocessing steps deployed for our model.

Problem formulation

NER in Urdu is an essential yet challenging task in NLP, necessitating the identification and classification of entities within text into predefined categories like locations, persons, organizations, etc. The challenge is amplified by Urdu's unique script and linguistic features, which influence every aspect of NER:

- **Entity Identification**: This involves detecting tokens within a sentence that correspond to named entities. The morphological richness of Urdu, which includes exten-

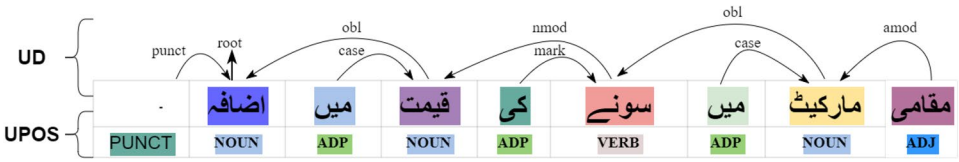

Fig. 2 An illustration for Universal Dependency and Part-of-speech tagging on a given example

	پاکستان	کے	وزیراعظم	عمران	خان	نے	اقوام	متحدہ	کی	جنرل	اسمبلی	میں	تقریر	کی
پاکستان	S-Loc	~	~	~	~	~	~	~	~	~	~	~	~	~
کے	~	O	~	~	~	~	~	~	~	~	~	~	~	~
وزیراعظم	~	~	B-Per	~	~	~	~	~	~	~	~	~	~	~
عمران	~	~	~	I-Per	~	~	~	~	~	~	~	~	~	~
خان	~	~	~	~	E-Per	~	~	~	~	~	~	~	~	~
نے	~	~	~	~	~	O	~	~	~	~	~	~	~	~
اقوام	~	~	~	~	~	~	B-Org	~	~	~	~	~	~	~
متحدہ	~	~	~	~	~	~	~	E-Org	~	~	~	~	~	~
کی	~	~	~	~	~	~	~	~	O	~	~	~	~	~
جنرل	~	~	~	~	~	~	~	~	~	B-Org	~	~	~	~
اسمبلی	~	~	~	~	~	~	~	~	~	~	E-Org	~	~	~
میں	~	~	~	~	~	~	~	~	~	~	~	O		~
تقریر	~	~	~	~	~	~	~	~	~	~	~	~	O	~
کی۔	~	~	~	~	~	~	~	~	~	~	~	~	~	O

Fig. 3 BIOES tagging matrix for the Urdu sentence

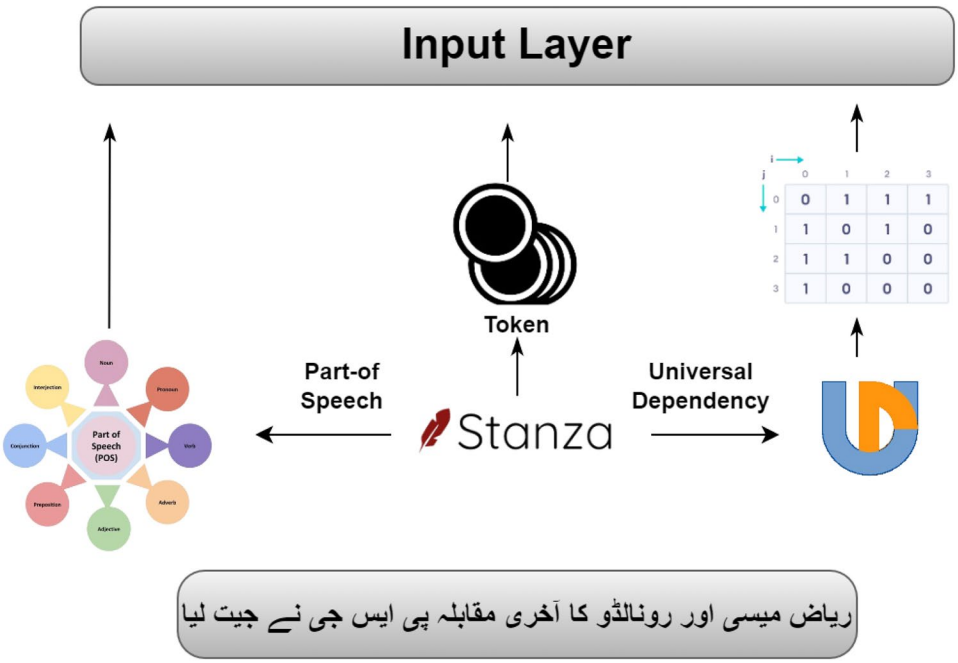

Fig. 4 An illustration of data preprocessing and features extraction

sive use of inflections, makes complex tokenization and the accurate determination of entity boundaries.

- **Entity Classification**: Classifying identified entities into categories such as Person, Location, Organization, Time, and Others. Complications arise from Urdu's script features, such as the absence of capital letters and extensive use of loanwords, which blur the distinction between common nouns and proper nouns.

Moreover, the flexibility in word order in Urdu syntax poses additional challenges in using positional cues for

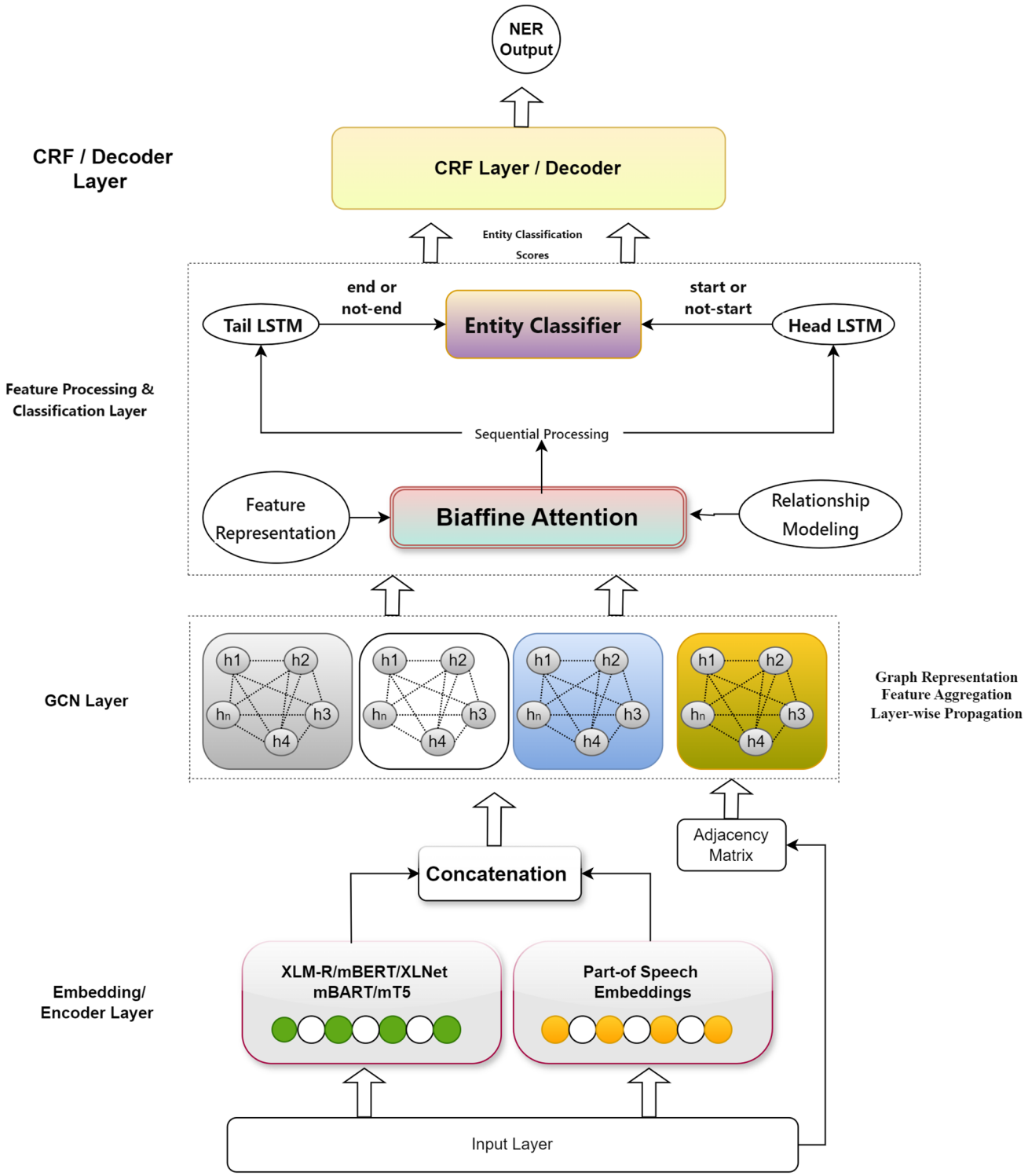

Fig. 5 The overall architecture of our Urdu Named-Entity Recognition model

entity recognition, which are typically relied upon in more rigidly structured languages.

Mathematical formulation of NER process

Dependency parsing and adjacency matrix

Dependency parsing in Urdu aims to depict syntactic dependencies that help in understanding relational structures within sentences, which is critical for accurate entity recognition an example has been presented in Fig. 2. Given a sentence S with tokens $t_1, t_2, \ldots, t_n$:

$$A[i][j] = \begin{cases} 1 & \text{if there is a syntactic dependency from } t_i \text{ to } t_j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

BIOES tagging scheme

The BIOES tagging scheme categorizes each token based on its position within an entity:

$$\text{Tag}(t_i) = \begin{cases} B-\text{type} & \text{if } t_i \text{ is the start of an entity} \\ I-\text{type} & \text{if } t_i \text{ is inside an entity} \\ E-\text{type} & \text{if } t_i \text{ is the end of an entity} \\ S-\text{type} & \text{if } t_i \text{ is a single-token entity} \\ O & \text{if } t_i \text{ is outside any entity} \end{cases}$$

For instance, in a morphologically complex sentence, the word بازار (bazaar) might appear with affixes that alter its appearance and syntactic role, potentially misleading simplistic tagging algorithms. Figure 3 depicted an example of implementing BIOES tagging on an example.

These refinements aim to give a clearer and more detailed account of how the NER model handles Urdu's linguistic complexities, providing a robust theoretical foundation for the model's design and expected functionalities.

Input representation

The input to the model is a sequence of tokens $S = \{t_1, t_2, ..., t_n\}$, where n is the length of the sentence. Each token t_i is first converted into a vector representation using a pre-trained XLM-R model and POS tag embeddings.

$$\mathbf{v}_i = \text{XLM-R}(t_i) \oplus \text{POS-Embed}(t_i) \tag{2}$$

Here, $\oplus$ denotes concatenation of the XLM-R output and POS tag embeddings for each token.

Graph convolutional network

The concatenated embeddings are then processed by a Graph Convolutional Network to capture dependency relations among the tokens. The GCN operates on the adjacency matrix A, computed from the dependency parse of the sentence.

$$\mathbf{h}^{(l+1)} = \text{ReLU}\left(\mathbf{A}\mathbf{h}^{(l)}\mathbf{W}^{(l)}\right) \tag{3}$$

Where $\mathbf{h}^{(l)}$ represents the hidden states at layer l, and $\mathbf{W}^{(l)}$ is the weight matrix for layer l.

Biaffine Attention

After GCN processing, the feature representations are enhanced using a Biaffine Attention mechanism, which models pairwise interactions between elements in the sequence to predict relationships and roles, crucial for entity recognition.

$$\mathbf{a}_{ij} = \mathbf{h}_i^T \mathbf{W}_b \mathbf{h}_j + \mathbf{u}^T \mathbf{h}_i + \mathbf{v}^T \mathbf{h}_j + b \tag{4}$$

Where $\mathbf{a}_{ij}$ is the attention score between tokens i and j, $\mathbf{W}_b$ is a bi-linear weight matrix, and $\mathbf{u}$, $\mathbf{v}$, and b are parameters for linear transformations and bias, respectively.

Feature aggregation

To employ the Biaffine Attention scores for entity classification, the model aggregates these 2D scores into 1D feature vectors. This can be achieved by summing or averaging the attention scores for each token, or by applying a pooling operation.

$$\mathbf{h}_i = \sum_j \mathbf{a}_{ij} \mathbf{h}_j \tag{5}$$

This aggregation step ensures that each token i has a corresponding feature vector $\mathbf{h}_i$ that incorporates the attention information from its interactions with all other tokens.

Entity classification

The computed feature vectors from the Biaffine Attention layer are then passed through a linear layer to classify each token into entity categories e.g., person, location, dates.

$$p(\mathbf{y}|\mathbf{x}) = \text{Softmax}\left(\mathbf{W}_c \mathbf{h} + \mathbf{b}_c\right) \tag{6}$$

Where $\mathbf{W}_c$ and $\mathbf{b}_c$ are the weights and biases of the classifier, $\mathbf{h}$ is the aggregated feature vector from the Biaffine Attention layer, and $\mathbf{y}$ represents the entity tags.

Conditional random field

Finally, a CRF layer is used over the output probabilities to ensure coherent label predictions across the sequence, leveraging the structured prediction capability.

$$P(\mathbf{Y}|\mathbf{X}) = \frac{\exp\left(\sum_{i=1}^{n} \mathbf{W}_{y_i,y_{i-1}} + \sum_{i=1}^{n} \mathbf{U}_{y_i}^T \mathbf{x}_i\right)}{\sum_{\mathbf{Y}' \in \mathcal{Y}(X)} \exp\left(\sum_{i=1}^{n} \mathbf{W}_{y'_i,y'_{i-1}} + \sum_{i=1}^{n} \mathbf{U}_{y'_i}^T \mathbf{x}_i\right)} \quad (7)$$

Where **W** and **U** are the transition and emission matrices in the CRF, respectively, **Y** is the sequence of predicted labels, **X** is the sequence of input feature vectors, and $\mathcal{Y}(X)$ represents all possible label sequences for **X**.

This formulation encapsulates the end-to-end process of the NER model from token representation to final entity prediction, employing advanced neural architectures and interactions for robust performance.

Model architecture

The overall architecture of our proposed Urdu NER model integrates multiple advanced components, each contributing to improved contextual understanding and entity detection. The model processes input Urdu sentences through the following key layers:

- **Embedding Layer:** Token embeddings are generated using pre-trained transformer-based models (XLM-R, mBERT, XLNet) and optionally combined with POS embeddings to capture both contextual and syntactic information.
- **Graph Convolutional Network (GCN):** Constructs a dependency graph using adjacency matrices to learn syntactic relations between words, refining token representations with structural information.
- **Biaffine Attention Mechanism:** Models pairwise token interactions, enhancing the model's ability to learn complex dependencies crucial for recognizing named entities.
- **BiLSTM-based Boundary Detection:** Captures sequential context to identify the start and end of entities in the text.
- **CRF Layer:** Enforces valid tag transitions using the BIOES scheme, improving the consistency of final predictions.

Figure 5 presents the complete end-to-end structure of the model.

Experiments

Evaluation criteria

To rigorously evaluate our NER model, we deployed standard metrics: Precision, Recall, and the F1 Score. Each of these metrics offers a unique perspective on the model's performance:

- **Precision:** Evaluate the accuracy of positive predictions. It is calculated as:

$$\text{Precision} = \frac{TP}{TP + FP}$$

- **Recall:** Calculate the model ability to identify all relevant instances of "True Positive" among all actual positives. It is defined as:

$$\text{Recall} = \frac{TP}{TP + FN}$$

- **F1 Score:** The harmonic mean of Precision and Recall, present a balance between them. It is specifically useful when the class distribution is imbalanced. The F1 Score is calculated as:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

In the context of NER, we evaluate our model at the entity level. This means that:

- A true positive is counted when the model correctly predicts an entire entity that matches the gold standard.
- A false positive is counted when the model predicts an entity that is not present in the gold standard.
- A false negative is counted when an entity present in the gold standard is not predicted by the model.

These metrics are crucial for understanding different aspects of the NER model's performance, and they allow for an objective comparison of its effectiveness. In our discussions, these formulas will be referenced to provide a comprehensive analysis of the model's capabilities in accurately categorizing and recognizing named entities within the dataset.

Computational complexity

To evaluate the efficiency and scalability of the proposed model, we analyze the computational complexity of its primary components. Let n denote the length of the input sequence, d the dimensionality of token embeddings, and L the number of possible entity labels.

- **Embedding Layer:** Generating contextual embeddings from transformer-based models (e.g., XLM-R) and POS embeddings involves linear complexity with respect to the sequence length, i.e., $O(n \cdot d)$.
- **Graph Convolutional Network (GCN):** Each GCN layer processes relational structures using adjacency matrices, resulting in a time complexity of $O(n^2 \cdot d)$, due

to the dense matrix multiplications between node features and the graph structure.

- **Biaffine Attention:** This layer calculates pairwise token interactions for dependency modeling, also incurring a complexity of $O(n^2 \cdot d)$, as each token attends to every other token in the sequence.
- **Entity Boundary Detection (BiLSTM):** The BiLSTM layer processes the sequence in both directions, with each step involving a matrix-vector operation, leading to a time complexity of $O(n \cdot d^2)$.
- **Conditional Random Field:** The CRF layer decodes the optimal tag sequence using dynamic programming with complexity $O(n \cdot L^2)$, where L is the number of entity labels.

Although certain components such as GCN and Biaffine Attention introduce quadratic complexity in terms of sequence length, the model remains computationally feasible due to efficient batching and parallelization on GPUs. This allows the architecture to scale effectively for real-time or near-real-time applications, even in the context of resource-constrained languages like Urdu.

Model configuration

Table 5 provides an overview of the parameters used in our extended model configuration.

The model takes advantage of deep learning transformer models embedding with a size of 768, which is standard for capturing the contextual representation of words in multiple languages. Similarly, POS tags are embedded with the same dimension to ensure that part-of-speech information is well-represented. The GCN with a layer size of 192 processes relational data between words, enhancing the model's ability to understand syntactic structures. Biaffine attention, which is particularly effective for relationship prediction in NLP tasks, is employed with a dimensionality of 17 to match the number of entity tags.

The model uses aa CRF layer to ensure that the predicted entity tags form valid sequences for NER tasks. Training is performed over 5 epochs with a batch size of 8, utilizing an Adam optimizer with a learning rate of 1×10^{-5}. The specified dropout rate of 0.5 helps prevent overfitting by randomly disabling neurons during training. Lastly, the model incorporates ReLU among other activation functions to introduce non-linearities into the learning process, which is crucial for deep learning models. The inference time for the model shows 100 ms, whereas the peak memory utilization was 27 GB.

Table 5 Model configuration parameters

Parameter	Setting
BERT Embedding Size	768
POS Tag Embedding Size	768
GCN Layer Size	192
Biaffine Attention Size	17
CRF Num Tags	17
Batch Size	8
Optimizer	Adam
Learning Rate	1×10^{-5}
Epochs	20
Total Trainable Parameters	178,797,669
Dropout Rate	0.5
Loss Function	CrossEntropyLoss
Activation Functions	ReLU and others
Inference Time	100 ms
Peak Memory Usage	27 GB

Dataset

Our Urdu NER dataset is carefully divided into train, dev, and test sets, as detailed in Table 6. This split makes sure a comprehensive approach to model training and performance evaluation. The majority of our dataset, specifically 80% or

Table 6 Detailed BIOES Tag Frequencies and Summary Statistics for Training, Development, and Test Sets

Category	Training	Test	Development
BIOES Tag Frequencies			
O	144,513	27,096	8,122
S-Location	2,207	413	124
S-Person	2,360	442	132
B-Person	4,713	883	264
E-Person	4,728	886	265
B-Time	340	64	19
E-Time	335	63	19
B-Location	687	129	38
E-Location	685	128	38
I-Person	902	169	50
S-Time	122	23	7
B-Organization	2,076	389	116
I-Organization	934	175	52
E-Organization	2,082	390	117
S-Organization	682	128	38
I-Time	15	3	1
I-Location	93	17	6
Summary Statistics			
Total Tokens	186,190	34,911	11,636
Total Entities	11,813	2,213	740
Organization	2,478	464	155
Person	6,307	1,182	395
Location	2,618	491	164
Time	411	77	26

186,190 tokens, is allocated to the training. This considerable collection is instrumental in training the NER model on a diverse array of examples that encapsulate the complex linguistic properties of Urdu, such as its diverse syntax and morphology. The remaining portions of the dataset, the development and test sets, are essential for refining and evaluating the model. The dev set, which includes 34,911 tokens, allows for iterative tuning of the model's parameters. This process helps optimize performance without compromising the coherence of the test data. The testing comprises 11,636 tokens and plays a critical role in the final evaluation. It ensures that the model performs reliably on novel data, thus demonstrating its ability to generalize beyond the training examples. This structured approach not only aids in rigorous training but also adheres to established machine learning protocols ensuring that evaluations are unbiased and that each dataset group serves a special purpose in developing a robust NER system. Each part of dataset is meticulously annotated with BIOES tags, setting entity boundaries and types, including persons, locations, times, and organizations, thereby furnishing a detailed framework for both the learning and assessment phases.

This labeling technique importantly escalates the performance and depth of our NER analysis, enabling this model to recognize the difference between detailed entity structures within the text. This kind of detailed approach to NER is key for the development of systems capable of grasping and interpreting nuanced linguistic constructs, particularly in diverse and complex datasets derived from varied news sources as in our study. These advanced NER capabilities are essential for applications ranging from sentiment analysis to automated content categorization and beyond, offering broad and detailed insights into the data's inherent linguistic and semantic structures.

Implementation details

Our NER model is tailored for processing Urdu news headlines, integrating advanced neural network architectures and linguistic features extracted from the dataset. The model architecture and training procedures are detailed below.

Dataset preparation

The dataset is a collection of manually annotated news headlines from different news channels. We used the Stanza NLP library to pre-process these headlines by extracting tokens, POS tags, lemmas, and syntactic dependencies. This pre-processing step makes sure rich input for training our neural networks.

As detailed in Table 7, pre-processing of news headlines involves leveraging the Stanza NLP library to meticulously prepare the data for analysis. This process includes tokenization, POS tagging, and syntactic dependency analysis. These steps are crucial for extracting meaningful linguistic features and facilitating accurate training of our deep learning models. This organized data preparation not only aids in improving the accuracy of tasks such as NER but also enhances the model's ability to interpret and analyze Urdu text effectively.

Training procedure

The model has been trained using an 80:15:5 split for train, test, and dev, respectively. This division aligns with best practices in machine learning, ensuring a well-balanced approach to evaluate the model while retaining sufficient data for effective training. This is designed to allow the model to generalize effectively, by exposing it to a wide variety of examples during the initial training phase, and eventually providing ample datasets for both assessing performance

Table 7 Detailed NLP Processing Results. For every item that is non-English in the paper, an accompanying English translation is provided

Headline Token	Token ID	Token	POS Tag	Dependency (Dep)	BIOES Tag
شیخ (Sheikh)	58210	شیخ	PROPN	(0,1)	B-Person
رشید (Rasheed)	167180	رشید	PROPN	(1,3)	E-Person
کی (of)	288	کی	ADP	(2,1)	O
رہائش (residence)	154896	رہائش	NOUN	(3,7)	O
گاہ (-place)	119898	گاہ	–	–	O
سے (from)	504	سے	ADP	(4,3)	O
سرکاری (official)	72878	سرکاری	ADJ	(5,6)	O
سیکیورٹی (security)	185421	سیکیورٹی	NOUN	(6,7)	O
ہٹا (removed)	189779	ہٹا	VERB	(7,7)	O
دی (given)	8598	دی	–	–	O
گئی (was)	10896	گئی	AUX	(8,7)	O

more rigorously during testing and fine-tuning during the development phase.

The training set comprises 186,190 tokens across 14,813 named entities, the test set includes 34,911 tokens with 2213 named entities, and the development set contains 11,636 tokens with 740 named entities. We utilized the Adam optimizer with a learning rate of 1×10^{-5} and trained the model for five epochs, making incremental adjustments based on test set performance to prevent over-fitting.

We can adopt the split ratio, such as 70:20:10 or 90:5:5 distribution, which could notably affect the model's evaluation and learning capabilities. If we consider a smaller train set, as in a 70:20:10 split, might lead to underfitting, where the model is unable to learn properly from the training data, affecting its ability to generalize. Whereas, increasing the size of the train set, as discussed in a 90:5:5 split, could affect the results of the test and development sets, potentially compromising the reliability of the performance metrics obtained during validation. Thus, the 80:15:5 split was chosen to optimize the balance between comprehensive training and robust validation and testing, ensuring reliable performance evaluations and model generalization.

Baselines

In this section, we evaluate our proposed model against various baseline models. For each baseline, we perform experiments utilizing mBERT, XLM-R, and FastText for the word embedding layer. Below is an overview of each baseline model used in our comparative analysis.

- **Haq et al.** [12] establish a comprehensive Urdu NER system that utilizes deep learning techniques to overcome the limitations of manual feature engineering. Their method integrates convolutional neural networks for character-level feature extraction with word embeddings, achieving a notable improvement in F1 score across four benchmark datasets.
- **Kanwal et al.** [23] propose a neural NER model that leverages recurrent architectures and diverse word embeddings such as fastText, Word2Vec, and GloVe. Their experiments report significant gains in Urdu NER performance, highlighting the potential of deep learning for this task.
- **Khan et al.** [14] introduce deep recurrent models enhanced by both language-specific and general linguistic features. Evaluated on three datasets, their approach surpasses traditional CRF and ANN-based methods, achieving F-measure scores up to 81.1%.
- **Ullah et al.** [49] present an Attention-BiLSTM-CRF framework tailored for Urdu NER using the MK-PUCIT Corpus. Their integration of word embeddings with a self-attention mechanism leads to a high F1-score of 92%, establishing a strong benchmark for Urdu-specific NER models.
- **Ullah et al. (CWEA)** [39] propose a contextual data augmentation strategy called Contextual Word Embeddings Augmentation (CWEA) for low-resource languages like Urdu. By enriching training datasets, they demonstrate how transformer-based models such as BERT and RoBERTa can achieve higher accuracy and robustness in complex linguistic environments.
- **Anam et al.** [38] develop a deep learning approach that utilizes FastText and Floret embeddings combined with LSTM, BiLSTM, GRU, and CRF architectures for Urdu NER. Their work demonstrates the effectiveness of contextual embeddings and recurrent architectures in improving NER accuracy and generalizability across multiple Urdu datasets.

Main results

The evaluation results of different models on the Urdu NER task have shown noteworthy variations in performance, underscoring how each model's architecture addresses the complex syntactic and morphological traits of Urdu. XLM-R outperforms its counterparts, achieving an F1-score of 98.69%, and an accuracy of 98.89%. These outcomes not only showcase the robust cross-lingual superiority of XLM-R but also its ability to navigate the intricate details of Urdu in predicting name entities, this could suggest that its architecture is especially aligned with the linguistic demands of the urdu language and its complexities.

Whereas, when we check the results for mBERT's F1-score of 90.05%, and accuracy of 90.94%, we see competent but not outstanding performance. This may be due to its generalized multilingual pre-training, which might not fully capture the unique linguistic challenges Urdu presents.

XLNet's evaluation results is moderate, indicating a precision of 92.65%, recall of 91.78%, F1-score of 92.30%, and an accuracy of 92.52%. Although XLNet's auto regressive training is favorable for some linguistic tasks, it is somewhat less effective for Urdu NER. This could be attributed to its sequential prediction focus, which may not adequately handle the broader contextual needs essential for effective NER.

While examining the encoder-decoder models mT5 and mBART demonstrate lower performance, with mT5 achieving a precision of 88.50%, recall of 89.00%, F1-score of 88.75%, and an accuracy of 89.20%, and mBART a precision of 87.90%, recall of 88.30%, F1-score of 88.10%, and an accuracy of 88.60%. Their underperformance could be linked to the dataset prominence on explicitly labeled entities, which is unable to exploit their strengths in generative capabilities that excel in contexts requiring the interpretation of implicit linguistic elements. These models might perform

Table 8 Comparative performance of different encoder models used in our NER system on Urdu NER

Encoder/Embeddings model	Precision	Recall	F1-Score	Accuracy
XLM-R	98.58	98.80	98.69	98.89
XLNet	92.65	91.78	92.30	92.52
mBERT	89.32	90.76	90.05	90.94
mT5 (Encoder)	88.50	89.00	88.75	89.20
mBART (Encoder)	87.90	88.30	88.10	88.60

better in scenarios that demand the generation of contextual embeddings and filling gaps based on subtle cues.

The evaluation emphasizes the need to align model architecture with the NER task's specific demands, especially in linguistically complicated settings like Urdu. XLM-R's superior results suggest its suitability for further specialized training or fine-tuning to boost NER performance in Urdu. Moreover, the potential improvements with mT5 and mBART in handling datasets enriched with implicit entities could catalyze eloquently in NER systems, utilizing their unique generative abilities to enhance the recognition of entities in linguistically challenging scenarios (see Table 8).

Model performance analysis

Model comparison

In the comparative evaluation presented in Table 9, the proposed model achieves the highest F1-score of 98.69%, outperforming all existing SOTA Urdu NER systems. This strong result is supported by a precision of 97.52% and a recall of 95.80%, reflecting the model's ability to maintain an optimal balance between accurate and complete entity identification. The incorporation of GCN with syntactic dependency parsing, Biaffine Attention for token-level interaction, and the robust contextual representation from XLM-R embeddings collectively contribute to this performance gain.

Among existing methods, Ullah et al. (CWEA 2024) achieved an impressive F1-score of 98.20% by leveraging BERT-multilingual embeddings and a contextual word embedding augmentation technique. Similarly, Anam et al. obtained an F1-score of 98.00% using a combination of BiLSTM-GRU with Floret embeddings, showcasing the strength of recurrent networks with contextualized embeddings.

While these models present competitive results, the proposed model surpasses them by integrating syntactic, positional, and contextual information more effectively, particularly suited for the morphological richness and free word order of Urdu. In contrast, models by Ullah et al. (2022), Haq et al., Khan et al., and Kanwal et al. fall behind in performance, largely due to either limited architectural depth or lack of advanced embedding strategies.

Table 9 Comparison of named entity recognition models performance

Model	Technique	Prec. (%)	Rec. (%)	F1 (%)
Haq et al. (2023) [12]	Bi-GRU + CNN	85.37	85.68	85.53
Kanwal et al (2019). [23]	RNN	76.30	78.90	77.50
Khan et al. (2022) [14]	LSTM-forward RNN	84.75	79.35	81.10
Ullah et al (2022). [49]	Attention Bi-LSTM + CRF	93.00	92.90	92.95
Ullah et al. (CWEA 2024) [39]	BERT-multilingual + Data Augmentation	97.90	98.40	98.20
Anam et al. [38]	BiLSTM + GRU + Floret	97.00	99.00	98.00
Proposed Model	GCN + CRF + XLM-R	97.52	95.80	98.69

Table 10 Performance Comparison of Named Entity Recognition Models for benchmark Datasets

Model	MK-PUCIT			IJCNLP			Jehangir et al.		
	P	R	F1	P	R	F1	P	R	F1
Haq et al.	–	–	–	66.67	67.13	66.90	83.85	83.60	83.73
Kanwal et al.	76.30	78.90	77.50	69.50	63.10	65.70	–	–	–
Khan et al.	–	–	–	61.20	70.01	63.21	79.71	81.56	79.94
Ullah et al.	91.30	93.30	92.30	–	–	-	–	–	–
Our Model	92.10	94.22	93.14	71.04	69.66	70.34	82.68	84.35	83.50

These findings demonstrate that the proposed model not only sets a new performance benchmark in Urdu NER but also confirms the efficacy of combining graph-based and transformer-based features for sequence labeling tasks in low-resource and morphologically complex languages.

Model analysis of results on benchmark datasets

In the comparative analysis of NER models, as detailed in Table 10, our Model presents a strong performance across multiple benchmark datasets, not just the custom dataset we specifically tailored for this research. This demonstrates the robustness and general applicability of our model in various contexts, a critical factor in the real-world deployment of NER systems.

For the MK-PUCIT and Jehangir et al. datasets, the Proposed Model achieved F1-scores of 93.14% and 83.50% respectively, surpassing the other models in the comparison. Particularly notable is its performance on the Jehangir et al. dataset, where it outperforms and shows marked improvements over Khan et al. and Haq et al. These results underline our model's superior ability to adapt to different linguistic features and complexities inherent in diverse Urdu datasets.

On the IJCNLP dataset, in spite of the fact that our model does not achieve the same dominance, it still provide a solid performance with an F1-score of 70.34%. This dataset, known for its varied text types and complex entity structures, presents a challenging environment where our model's precision of 71.04% and recall of 69.66% are competitive, particularly when compared against the performances of Haq et al. and Khan et al.

The consistent performance across different datasets, particularly in comparison to established models by Ullah et al., Khan et al., and others, emphasizes the effectiveness of our proposed model in handling the nuanced demands of Urdu NER. The systematic advances in model accuracy, as evidenced by these comparative results, reinforce the effectiveness of the advanced algorithms and optimization techniques employed in our model, which are specifically tailored to enhance precision, recall, and overall efficiency in real-world NER applications.

Model analysis

In the Fig. 6 presented, the comparison of training loss between mBERT, XLM-R, and XLNet over 20 epochs shows distinct learning patterns for each model. mBERT exhibits a steady decrease in training loss, indicating consistent learning throughout the epochs. In contrast, XLM-R shows a rapid decline in loss early on, stabilizing at a low level, suggesting that it quickly grasps the essential patterns in the data. XLNet starts with the highest loss but rapidly improves, reaching a low and stable loss after about 10 epochs, with minor fluctuations thereafter, indicating effective learning without overfitting. This comparison highlights the varying efficiencies and dynamics of each model in processing and learning from the same dataset.

Entity recognition performance analysis

The enhanced visualization of the confusion matrix in Fig. 7for entity types such as Organization, Person, Location, and Time provides clear insights into the model's performance in identifying named entities within Urdu news headlines. With True Positives peaking at 1882 for Persons and as low as 127 for Time, the model demonstrates a robust ability to recognize personal names, which are often distinct and frequent within text data. However, the lower TP for Time highlights challenges in identifying temporal

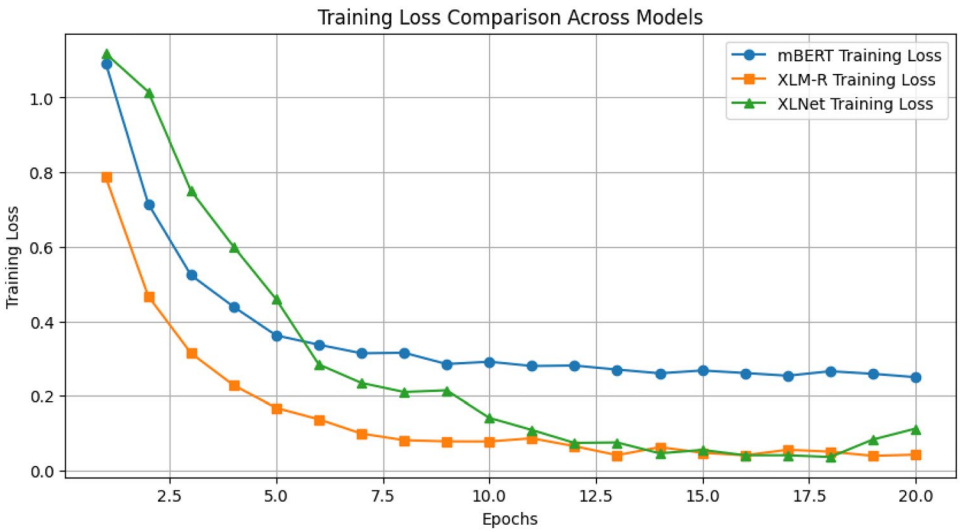

Fig. 6 Training Loss Comparison For XLM-R, XLNET and multilingual BERT Integrated

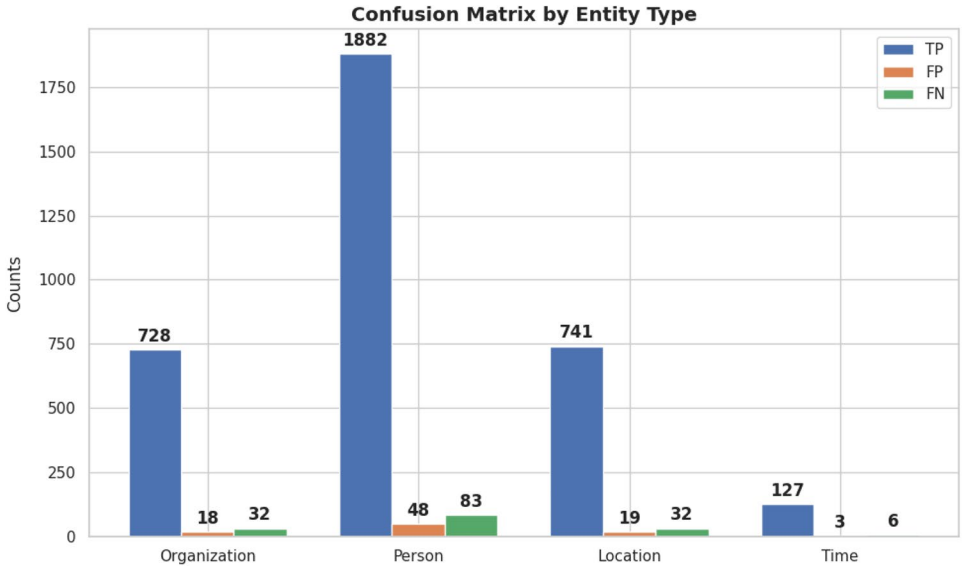

Fig. 7 Confusion matrix results of model while using XLM-R embeddings

Table 11 Ablation study

Model	Precision	Recall	F1-Score
Proposed Model	97.52	95.80	98.69
w/o GCN Layer	92.00	89.78	90.87
w/o CRF Layer	91.01	92.00	91.50
w/o GCN & CRF	86.45	85.92	86.18

expressions, potentially due to their varied formats and contextual usage in Urdu. The model maintains commendable precision, as seen with the highest False Positives at 48 for Persons, suggesting it occasionally misclassifies non-entities as persons. Meanwhile, False Negatives are more common for Persons and Locations, indicating recall issues where the model fails to identify all relevant entities, likely due to complex entity boundaries and subtle linguistic cues in the data.

Consider this Urdu news headline وزیراعظم عمران خان نے اسلام آباد میں قومی اسمبلی کے اجلاس میں شرکت کی۔ Prime Minister Imran Khan attended a session of the National Assembly in Islamabad. In this instance, the model successfully identifies "وزیراعظم عمران خان" as a Person and "اسلام آباد" as a Location, showcasing its high accuracy in recognizing well-known names. However, suppose the model incorrectly tags "قومی اسمبلی" (National Assembly) as an Organization rather than as a Location. This error could be attributed to the dual nature of the term, which can be perceived as both an organization and a location, reflecting the model's challenges in handling entities that can belong to multiple categories depending on the context. This example highlights the model's strengths in identifying clear-cut names but also underscores the complexities involved in entity recognition where contextual nuances play a significant role.

Ablation Study

In the ablation study presented in Table 11, we evaluate the individual and combined contributions of the GCN and CRF layers in our Urdu NER model. The full model configuration, integrating XLM-R embeddings, GCN for syntactic relation modeling, and CRF for tag sequence optimization, achieves the highest performance with an F1-score of 98.69%. When the GCN layer is removed, the F1-score drops to 90.87%, highlighting its importance in capturing dependency-based token relationships, which are especially critical in low-resource and morphologically rich languages like Urdu.

Similarly, eliminating the CRF layer results in an F1-score of 91.50%, showing that while contextual and structural information is still leveraged, the model loses effectiveness in maintaining coherent tag sequences. When both the GCN and CRF layers are removed, performance drops further to 86.18%, demonstrating the compound impact of losing both syntactic structure and sequence dependency modeling. This confirms that while transformer embeddings like XLM-R form a strong foundation, the integration of GCN and CRF layers is essential to enhance entity boundary detection, structural consistency, and overall robustness of the model.

Case Study

The case study presented in Table 12 offers a comparative analysis of several SOTA model embeddings mBERT, XLM-R, XLNet, mBART, and mT5 on NER in Urdu news headlines. Each model utilizes distinct embeddings, such as the multilingual BERT embeddings for mBERT and XLM-R, permutation-based embeddings for XLNet, and sequence-to-sequence embeddings for mBART and mT5.

Table 12 Case study: model comparison on Urdu news headlines with aspect entities

Models	Reviews
Review 1: "وزیر خزانہ نے 2024 کے بجٹ کی تفصیلات بیان کی"	
"Minister of Finance presented the details of the 2024 budget"	
Ground Truth	[Person: وزیر خزانہ (Minister of Finance), Organization: حکومت (Government)]
mBERT	[Person: وزیر خزانہ (Minister of Finance) ✓, Organization: حکومت (Government)] ✓
XLM-R	[Person: وزیر خزانہ (Minister of Finance) ✓, Organization: حکومت (Government)] ✓
XLNet	[Person: وزیر خزانہ (Minister of Finance) ✓, Organization: حکومت (Government)] ✓
mBART	[Person: وزیر خزانہ (Minister of Finance) ✓, Organization: حکومت (Government)] ✓
mT5	[Person: وزیر خزانہ (Minister of Finance) ✓, Organization: حکومت (Government)] ✓
Review 2: "پاکستان اور بھارت کے وزرائے خارجہ کی ملاقات"	
"Meeting of the Foreign Ministers of Pakistan and India"	
Ground Truth	[Person: وزیر خارجہ (Foreign Minister), Location: پاکستان (Pakistan), Location: بھارت (India), Organization: وزارت خارجہ (Ministry of Foreign Affairs)]
mBERT	[Person: وزیر خارجہ (Foreign Minister) ✓, Location: پاکستان (Pakistan) ✓, Location: بھارت (India) ✓, Organization: وزارت خارجہ (Ministry of Foreign Affairs)] ✓
XLM-R	[Person: وزیر خارجہ (Foreign Minister) ✓, Location: پاکستان (Pakistan) ✓, Location: بھارت (India) ✓, Organization: وزارت خارجہ (Ministry of Foreign Affairs)] ✓
XLNet	[Person: وزیر خارجہ (Foreign Minister) ✓, Location: پاکستان (Pakistan) ✓, Location: بھارت (India) ✓, Organization: وزارت خارجہ (Ministry of Foreign Affairs)] ✓
mBART	[Person: وزیر خارجہ (Foreign Minister) ✓, Location: پاکستان (Pakistan) ✓, Location: بھارت (India) ✓, Organization: وزارت خارجہ (Ministry of Foreign Affairs)] ✓
mT5	[Person: وزیر خارجہ (Foreign Minister) ✓, Location: پاکستان (Pakistan) ✓, Location: بھارت (India) ✓, Organization: وزارت خارجہ (Ministry of Foreign Affairs)] ✓
Review 3: "پاکستان میں پی ٹی آئی نے نیا سیاسی جلسہ منعقد کیا"	
"A new political rally was held by PTI in Pakistan"	
Ground Truth	[Location: پاکستان (Pakistan), Organization: پی ٹی آئی (PTI), Implicit Organization: سیاسی جماعت (Political Party)]
mBERT	[Location: پاکستان (Pakistan) ✓, Organization: پی ٹی آئی (PTI) ✓ , Implicit Organization: سیاسی (Political) ×
XLM-R	[Location: پاکستان (Pakistan) ✓, Organization: پی ٹی آئی (PTI) ✓ , Implicit Organization: جماعت (Party) ×
XLNet	[Location: پاکستان (Pakistan) ✓, Organization: پی ٹی آئی (PTI) ✓ , Implicit Organization: سیاسی (Political) ×
mBART	[Location: پاکستان (Pakistan) ✓, Organization: پی ٹی آئی (PTI) ✓ , Implicit Organization: سیاسی جماعت (Political Party) ✓
mT5	[Location: پاکستان (Pakistan) ✓, Organization: پی ٹی آئی (PTI) ✓ , Implicit Organization: سیاسی جماعت (Political Party) ✓

The results show that while all models performed well in identifying person and organization entities, mBERT, XLM-R, and XLNet struggled with implicit organizations. In contrast, mBART and mT5 demonstrated superior performance in detecting implicit organizations. These findings emphasize the impact of model embeddings on entity recognition tasks, providing a basis for improving NER systems in Urdu.

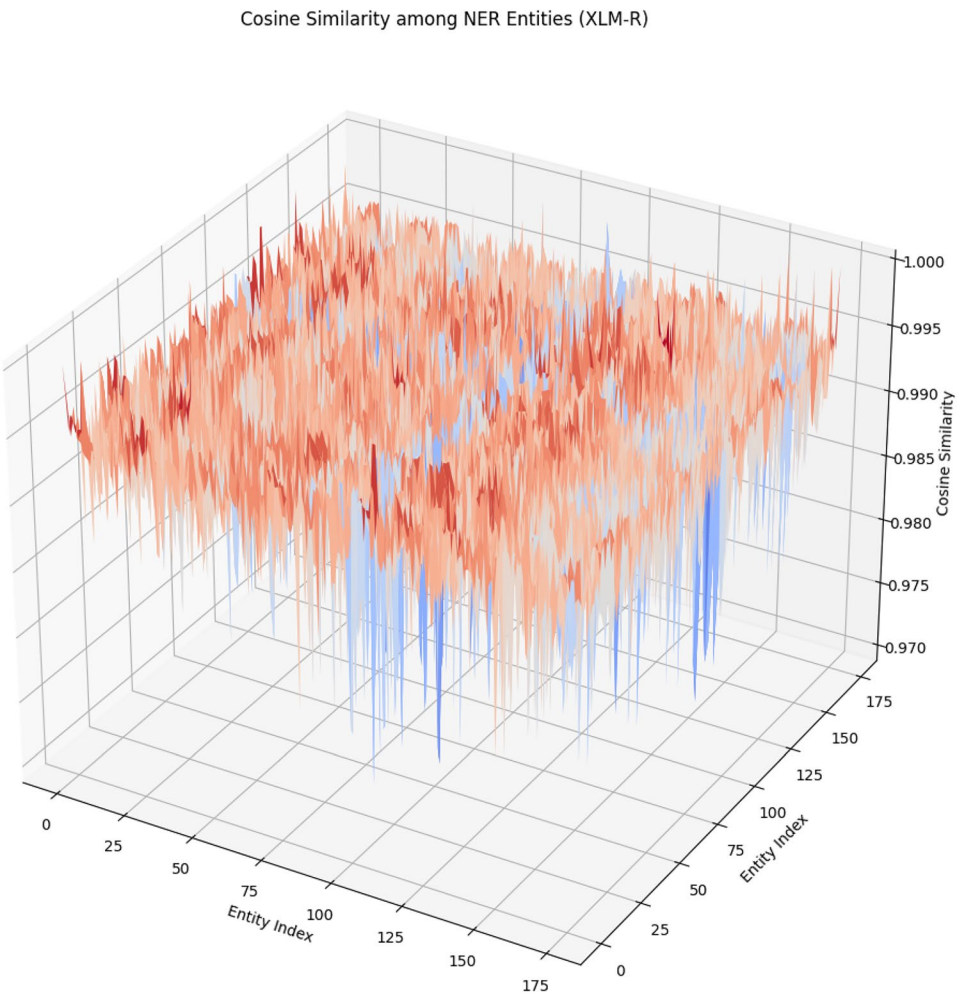

Fig. 8 Cosine similarity among NER entities using XLM-R

Analysis of word embeddings

The cosine similarity among entities using the XLM-R model Fig. 8 shows high uniformity, with values clustering between 0.98 and 1.00. This indicates that XLM-R produces stable and uniform embeddings for entities, which is superior for reliable entity recognition. The smooth distribution of similarity values underscores XLM-R's capability to maintain uniformity across embeddings, proving it the best choice for generating high-quality embeddings for Urdu text.

Using XLNet Fig. 9, the cosine similarity remains high but shows more variation compared to XLM-R, with values ranging from 0.92 to 1.00. This suggests XLNet captures finer nuances and distinctions among entities, enhancing detailed contextual understanding. Even though not being inherently multilingual, XLNet's robust contextual learning allows it to perform well in NER tasks for languages like Urdu. This robustness can be attributed to its autoregressive architecture, which excels in capturing dynamic contextual embeddings.

In contrast mBERT Fig. 10 displays a wide range of similarity values from 0.3 to 1.0, indicating variability. This suggests multilingual BERT captures diverse entity characteristics but with less stability, impacting consistency in NER tasks. The striking dips in similarity scores reflect mBERT's sensitivity to different entities, resulting in less stable embeddings as compared to XLM-R and XLNet.

In summary, XLM-R proves to be the most stable and uniform embeddings, making it the best choice for tasks requiring consistent and reliable entity representations. XLNet, although not multilingual, performs better than Multilingual BERT due to its ability to capture detailed nuances through its autoregressive architecture. BERT, despite its multilingual training, shows greater variability, making it less effective for tasks requiring stable embeddings. The choice of model should be guided by the specific requirements of the

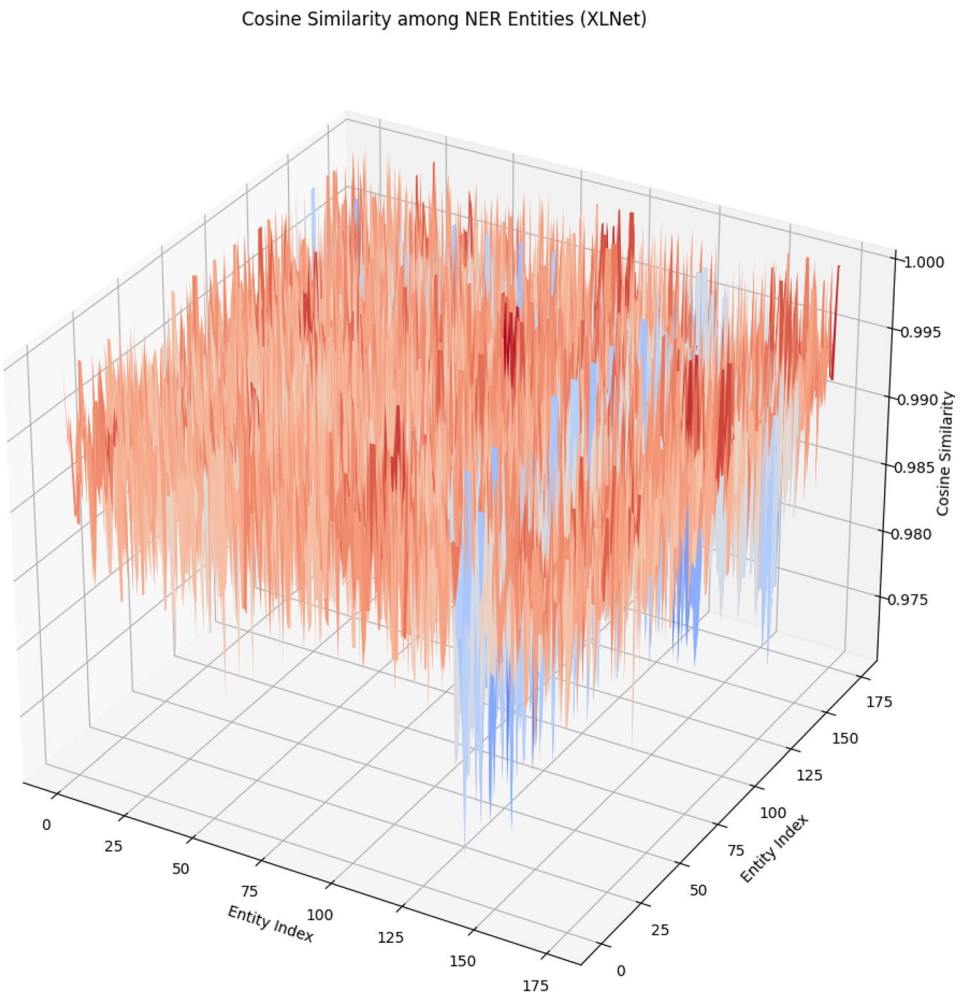

Fig. 9 Cosine Similarity among NER Entities using XLNet

NER task, balancing the need for embedding consistency and contextual nuance.

Conclusion

This research presents a novel Urdu NER system, optimized for processing news headlines. To address the considerable gap in resources for Urdu NER, we curated a specialized corpus that reflects the orthographic and morphological complexities distinctive to the Urdu language. Our model integrates advanced deep learning neural network technologies such as transformers, GCN, Biaffine Attention mechanisms, and a CRF layer, all of which work in concert to enhance word representation and entity recognition. Importantly, the inclusion of encoder-decoder models like mBART and mT5 has expanded the model's capability, particularly in handling implicit entity recognition and generating nuanced contextual embeddings. These models leverage their generative capabilities to fill linguistic gaps, offering substantial improvements in accurately identifying and labeling entities that are not explicitly mentioned but implied within the text. The robust performance of our system establishes a new standard in Urdu NER, markedly elevating the accuracy of entity recognition. The system adeptness at capturing the subtle nuances of the Urdu language and effectively managing the complexities of news headlines marks a notable improvement in computational linguistics for Urdu. This progress promises to enhance several applications, including automated content categorization, news aggregation, and advanced search functionalities within Urdu digital platforms. As NLP technology move forward, contributions from our research study provide a basis for more varied, inclusive, and comprehensive perspectives to understanding languages like Urdu.

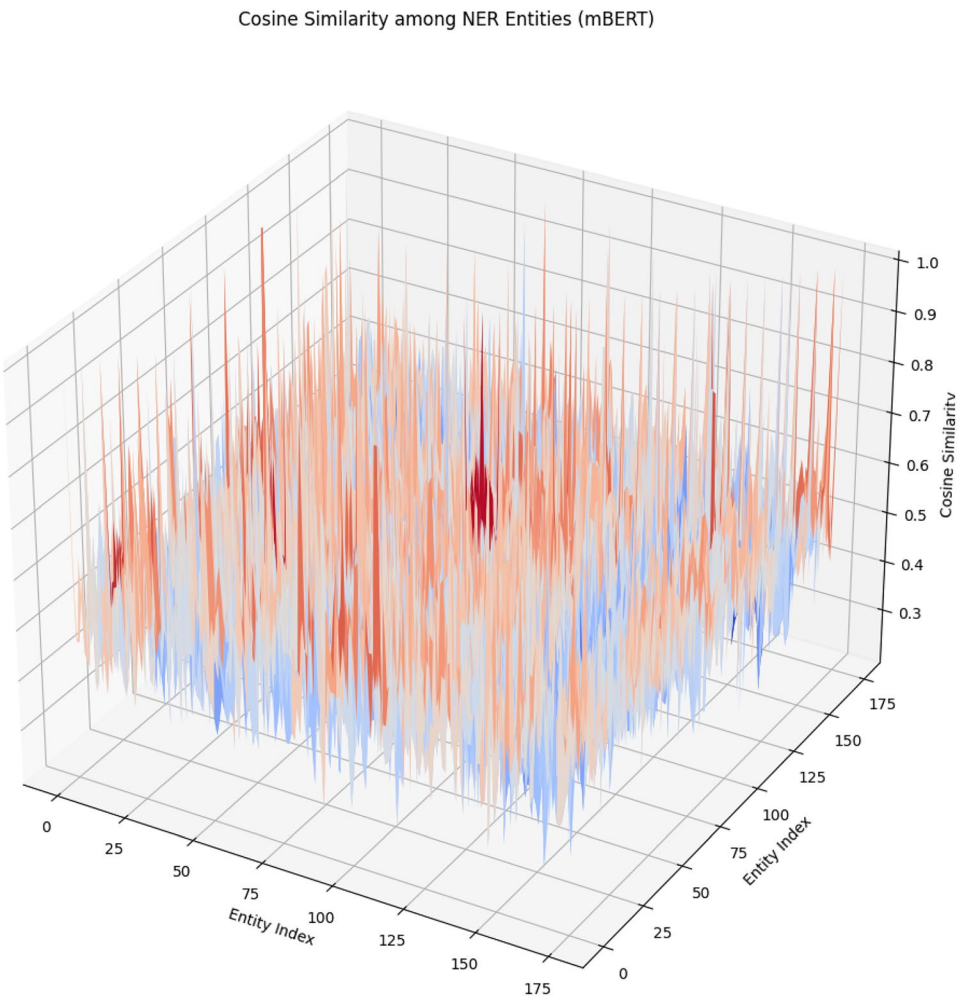

Fig. 10 Cosine Similarity among NER Entities using mBERT

Limitations and future work

While the proposed Urdu NER model shows strong performance on benchmark datasets, it has some limitations. The use of transformer-based architectures like mT5 and mBART introduces computational overhead, restricting deployment in resource-constrained environments and limiting real-time or mobile applications.

Additionally, the model is trained on a domain-specific dataset of Urdu news headlines, which may hinder its generalization to other textual genres such as literary prose, social media, or conversational Urdu, where language use is more informal and varied.

To address these limitations, future work will include expanding the training corpus to cover social media, literature, and user-generated content to improve generalization across different forms of Urdu text. Testing on similar languages like Arabic and Farsi will also be explored to assess cross-linguistic generalizability. Furthermore, the dataset will be extended to include implicit and nested entities, and efficiency benchmarks (inference time, memory usage) will be introduced to evaluate scalability for real-world deployment.

Acknowledgment The authors would like to acknowledge the support of Prince Sultan University, Saudi Arabia in paying the Article Processing Charges (APC) for this publication. The final manuscript was read and approved by all listed authors.

Data Availability The dataset generated and analyzed during this study will be made available upon request. It is intended to be released publicly following an embargo period or at a designated future date via the following GitHub repository (https://github.com/urdusentimentanalysis/URDUABSA).

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Yu J, Bohnet B, Poesio M (2020) Named entity recognition as dependency parsing. CoRR. arXiv:2005.07150
2. Song B, Li F, Liu Y, Zeng X (2021) Deep learning methods for biomedical named entity recognition: a survey and qualitative comparison. Brief Bioinform 22:bbab282, https://doi.org/10.1093/bib/bbab282. arXiv:https://academic.oup.com/bib/article-pdf/22/6/bbab282/41089553/bbab282.pdf
3. Li X, Zhang H, Zhou X-H (2020) Chinese clinical named entity recognition with variant neural structures based on bert methods. J Biomed Inform 107:103422. https://doi.org/10.1016/j.jbi.2020.103422
4. Liu Z et al (2021) Crossner: evaluating cross-domain named entity recognition. In: Proceedings of the AAAI Conference on artificial intelligence 35:13452–13460. https://doi.org/10.1609/aaai.v35i15.17587
5. Wang J, Shou L, Chen K, Chen G Pyramid: a layered model for nested named entity recognition. In: Jurafsky D, Chai J, Schluter N, Tetreault J (eds) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 5918–5928, https://doi.org/10.18653/v1/2020.acl-main.525 (Association for Computational Linguistics, Online, 2020)
6. Huang J et al (2021) Few-shot named entity recognition: an empirical baseline study. In: Moens M-F, Huang X, Specia L, Yih SW-t (eds) Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp 10408–10423, https://doi.org/10.18653/v1/2021.emnlp-main.813 (Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2021)
7. Ma J et al (2022) Label semantics for few shot named entity recognition. arXiv:2203.08985
8. Maham S et al (2024) Ann: adversarial news net for robust fake news classification. Sci Rep 14:7897
9. Riaz K (2010) Rule-based named entity recognition in Urdu. In: Proceedings of the 2010 Named Entities Workshop, pp 126–135
10. Lin BY, et al (2020) Triggerner: learning with entity triggers as explanations for named entity recognition. CoRR **abs/2004.07493**. arXiv:2004.07493
11. Riaz F, Anwar MW, Muqades H (2020) Maximum entropy based Urdu named entity recognition. In: 2020 International Conference on Engineering and Emerging Technologies (ICEET), pp 1–5, https://doi.org/10.1109/ICEET48479.2020.9048203
12. Haq R, Zhang X, Khan W, Feng Z (2023) Urdu named entity recognition system using deep learning approaches. Comput J 66:1856–1869
13. Aziz K et al (2024) Urduaspectnet: fusing transformers and dual gcn for urdu aspect-based sentiment detection. ACM Trans Asian Low-Resour Lang Inf Process. https://doi.org/10.1145/3663367. Just Accepted
14. Khan W, Daud A, Alotaibi F, Aljohani N, Arafat S (2020) Deep recurrent neural networks with word embeddings for Urdu named entity recognition. ETRI J 42:90–100
15. Khana W, Daudb A, Nasira JA, Amjada T (2016) Named entity dataset for urdu named entity recognition task. Lang Technol 51
16. Jahangir F, Anwar W, Bajwa UI, Wang X (2012) N-gram and gazetteer list based named entity recognition for Urdu: a scarce resourced language. In: Proceedings of the 10th Workshop on Asian Language Resources, pp 95–104
17. Tahir MS, Amjad M, Ahmad M, Ikram M, Fazal N (2024) Named entity recognition for Urdu language. Remit Rev 9:2724–2732
18. Al-Smadi M, Al-Zboon S, Jararweh Y, Juola P (2020) Transfer learning for Arabic named entity recognition with deep neural networks. IEEE Access 8:37736–37745. https://doi.org/10.1109/ACCESS.2020.2973319
19. He B, Chen J (2021) Named entity recognition method in network security domain based on bert-bilstm-crf. In: 2021 IEEE 21st International Conference on Communication Technology (ICCT), 508–512, https://doi.org/10.1109/ICCT52962.2021.9657857
20. Jarrar M, Khalilia M, Ghanem S (2022) Wojood: nested Arabic named entity corpus and recognition using bert. arXiv preprint arXiv:2205.09651
21. Zhou J et al (2021) Generative sentiment analysis via latent category distribution and constrained decoding. In: International Conference on artificial neural networks, pp 209–223 (Springer)
22. Malik MK (2017) Urdu named entity recognition and classification system using artificial neural network. ACM Trans Asian Low-Resour Lang Inf Process. https://doi.org/10.1145/3129290
23. Kanwal S, Malik K, Shahzad K, Aslam F, Nawaz Z (2019) Urdu named entity recognition: Corpus generation and deep learning applications. ACM Trans Asian Low-Resour Lang Inf Process. https://doi.org/10.1145/3329710
24. Malmasi S, Fang A, Fetahu B, Kar S, Rokhlenko O (2022) Multiconer: a large-scale multilingual dataset for complex named entity recognition. arXiv:2208.14536
25. Mahlous AR (2024) The impact of fake news on social media users during the covid-19 pandemic, health, political and religious conflicts: A deep look. Int J Psychol Relig 5:481–492
26. Dutta AK, et al (2023) Optimal weighted extreme learning machine for cybersecurity fake news classification. Comput Syst Sci Eng 44
27. Aziz K et al (2024) Unifying aspect-based sentiment analysis bert and multi-layered graph convolutional networks for comprehensive sentiment dissection. Sci Rep 14:14646
28. Yusufu A et al (2024) Uzbek news corpus for named entity recognition. In: Language resources and evaluation, pp 1–14
29. Riaz K (2010) Rule-based named entity recognition in Urdu. In: Kumaran A, Li H (eds) Proceedings of the 2010 Named Entities Workshop, 126–135 (Association for Computational Linguistics, Uppsala, Sweden, 2010)
30. Becker D, Riaz K (2002) A study in Urdu corpus construction. In: COLING-02: The 3rd Workshop on Asian Language Resources and International Standardization
31. Singh U, Goyal V, Lehal GS (2012) Named entity recognition system for Urdu. In: Kay M, Boitet C (eds) Proceedings of COLING, 2507–2518 (The COLING 2012 Organizing Committee, Mumbai, India, 2012)
32. Etzioni O et al (2005) Unsupervised named entity extraction from the web: an experimental study. Artif Intell 165:91–134. https://doi.org/10.1016/j.artint.2005.03.001
33. Sekine S, Nobata C (2004) Definition, dictionaries and tagger for extended named entity hierarchy. In: Proceedings of the Language Resources and Evaluation Conference (LREC), 1977–1980 (Lisbon, Portugal, 2004)
34. Jahangir F, Anwar W, Bajwa UI, Wang X (2012) N-gram and gazetteer list based named entity recognition for Urdu: a scarce resourced language. In: Weerasinghe R, Hussain S, Sornlertlamvanich V, Roxas REO (eds) Proceedings of the 10th Workshop on Asian Language Resources, 95–104 (The COLING 2012 Organizing Committee, Mumbai, India, 2012)
35. Malik MK, Sarwar SM (2017) Urdu named entity recognition system using hidden Markov model. Pak J Eng Appl Sci

36. Riaz F, Anwar MW, Muqades H (2020) Maximum entropy based Urdu named entity recognition. In: 2020 International Conference on Engineering and Emerging Technologies (ICEET), 1–5, https://doi.org/10.1109/ICEET48479.2020.9048203
37. Khan W et al (2022) Named entity recognition using conditional random fields. Appl Sci. https://doi.org/10.3390/app12136391
38. Anam R et al (2024) A deep learning approach for named entity recognition in Urdu language. PLoS ONE 19:e0300725
39. Ullah F, Gelbukh A, Zamir MT, Rivern EMF, Sidorov G (2024) Enhancement of named entity recognition in low-resource languages with data augmentation and bert models: a case study on urdu. Computers 13:258
40. Ullah F, Ullah I, Kolesnikova O (2022) Urdu named entity recognition with attention bi-lstm-crf model. In: Pichardo Lagunas O, Martínez-Miranda J, Martínez Seis B (eds) Advances in computational intelligence. Springer Nature Switzerland, Cham, pp 3–17
41. Biswas S, Mohanty S, Mishra SP (2009) A hybrid oriya named entity recognition system: Integrating hmm with maxent. In: 2009 Second International Conference on Emerging Trends in Engineering & Technology, 639–643 (IEEE, 2009)
42. Saha SK, Ghosh PS, Sarkar S, Mitra P (2008) Named entity recognition in Hindi using maximum entropy and transliteration. Polibits 38:33–41
43. Gali K, Surana H, Vaidya A, Shishtla PM, Sharma DM (2008) Aggregating machine learning and rule based heuristics for named entity recognition. In: Proceedings of the IJCNLP-08 Workshop on Named Entity Recognition for South and South East Asian Languages
44. Bikel DM, Miller S, Schwartz R, Weischedel R (1998) Nymble: a high-performance learning name-finder. arXiv preprint cmp-lg/9803003
45. Kumar P, Kiran VR (2008) A hybrid named entity recognition system for south Asian languages. In: Proceedings of The IJCNLP-08 Workshop on NER For South And South East Asian Languages, 83–88
46. Chaudhuri BB, Bhattacharya S (2008) An experiment on automatic detection of named entities in Bangla. In: Proceedings of the IJCNLP-08 Workshop on Named Entity Recognition for South and South East Asian Languages
47. Aziz K et al (2024) Enhanced urduaspectnet: leveraging biaffine attention for superior aspect-based sentiment analysis. J King Saud Univ-Comput Inform Sci 36:102221
48. Aziz K, Ji D, Li B, Li F, Zhou J (2024) Advancing Urdu nlp: aspect-based sentiment analysis with graph attention networks. In: 2024 International Joint Conference on Neural Networks (IJCNN), 1–8 (IEEE, 2024)
49. Ullah F, Ullah I, Kolesnikova O (2022) Urdu named entity recognition with attention bi-lstm-crf model. In: Mexican International Conference on Artificial Intelligence, 3–17 (Springer, 2022)
50. Nawaz A et al (2020) Extractive text summarization models for Urdu language. Inform Process Manag 57:102383. https://doi.org/10.1016/j.ipm.2020.102383
51. Sang EFTK, Meulder FD (2003) Introduction to the conll-2003 shared task: Language-independent named entity recognition. CoRR **cs.CL/0306050**
52. Ji H, Grishman R (2008) Refining event extraction through cross-document inference. In: Proceedings of ACL-08: Hlt, 254–262
53. Kirkpatrick A (2010) Researching English as a lingua franca in Asia: The Asian corpus of English (ace) project. Asian Eng 13:4–18
54. Zhou G, Su J (2005) Machine learning-based named entity recognition via effective integration of various evidences. Nat Lang Eng 11:189–206
55. Chinchor NA (1998) Overview of MUC-7. In: Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29 - May 1, 1998
56. Delgado R, Tibau X-A (2019) Why Cohen's kappa should be avoided as performance measure in classification. PLoS ONE 14:e0222916
57. Kvålseth TO (1989) Note on Cohen's kappa. Psychol Rep 65:223–226
58. Blackman NJ-M, Koval JJ (2000) Interval estimation for Cohen's kappa as a measure of agreement. Stat Med 19:723–741