A Deterministic Finite-State Morphological Analyzer for Urdu Nominal System

Abdulaziz Alblwi
Department of Computer Science, Applied College, Taibah University, Saudi Arabia
ablwi@taibahu.edu.sa

Mohammad Mahyoob
Department of Languages and Translation, Faculty of Science and Arts, Taibah University, Saudi Arabia
mqassem@taibahu.edu.sa (corresponding author)

Jeehaan Algaraady
Centre of Languages, Taiz University, Yemen
jihan.amu@gmail.com

Khateeb Syed Mustafa
Department of Linguistics, Aligarh Muslim University, India
khateebsyedmustafa@gmail.com

Received: 28 February 2023 | Revised: 29 April 2023 | Accepted: 2 May 2023

Licensed under a CC-BY 4.0 license | Copyright (c) by the authors | *DOI: https://doi.org/10.48084/etasr.5823*

ABSTRACT

The morphological analyzer is a computational process that combines lemmas with other linguistic features to produce new lexical word forms. This paper investigates the processing of a nominal system in the Urdu language. It focuses on the inflections of noun forms and studies number, gender, person, and case representations, using a Finite State Machine (FSM) to analyze and create all the possible forms of the standardized registers. The application of the analysis using this tool provides and displays all the possible structures and their declensions. This study adds all the necessary features and values to the lexical concatenating nouns according to their patterns. The accuracy score of the output is 92.7, where the actual output depends on the detailed design of the FSM and the specific morphological processes provided to the finite state tools.

Keywords-computational morphology; Urdu language; morphological analyzer; finite-state automata; inflection; derivation

I. INTRODUCTION

Hindustani is the lingua franca of India and Pakistan, which was developed into two standardized registers: Hindi and Urdu. The main difference between these two registers is the use of different scripts [1]. For example, Hindi uses the Devanagari script, while Urdu uses an extended form of the Perso-Arabic script. Urdu, Kaithi and Devanagari scripts are indigenous to India. The Urdu script is primarily used to write the Urdu language, while the Kaithi script is used for various languages, including Bhojpuri, Maithili, and Magahi. On the other hand, the Devanagari script is used for several languages, including Hindi, Marathi, Nepali, and Sanskrit. It is important to note that these scripts have evolved and have been influenced by various other scripts, such as Arabic, Persian, and Sanskrit. These scripts' continued use and preservation are crucial in maintaining India's diverse linguistic and cultural identity. In Urdu grammar, nouns are classified into two genders: masculine and feminine. This classification is based on the natural gender of the object being referred to. The number of nouns in Urdu can be singular or plural. The plural form of a noun in Urdu is formed by adding a suffix to the singular form. This system of gender and number classification is an essential aspect of the Urdu language and is used extensively in its literature and everyday communication [2].

A. Urdu Morphology

It is suggested that Hindi and Urdu share 82% of their daily vocabulary [1, 3], but this number drops if we move to a more specific domain to 69% for tourist phrases. "Hindi" is a very loose term covering widely varying dialects. It has been used to develop multilingual grammars that can be used for translation. Authors in [3] demonstrated that Hindi and Urdu share a grammar, but the lexicons diverge hugely beyond basic and general registers. Telugu/Kannada uses a Sanskrit-based lexicon essentially identical to that of Hindi. Morphology

studies how words are formed out of smaller units called morphemes [4]. The various morphological processes employed in Urdu are: (i). Affixation, (ii) zero-modification, (iii) reduplication, (iv) internal-change, and (v) suppletion. Morphology is about the structure of words and is mainly concerned with the formation of words in human languages [5]. Urdu has many examples in which certain words are formed by repetitions of the same form. Bloomfield referred to inflection as the outer layer of the morphology of word forms and derivation as the inner layer. This inflectional affixation is not used to produce new words in Urdu, they are used only to indicate aspects of the grammatical function of a word. The Urdu post-positions: /par/, /se/, and /ne/ may be grouped in locative case markers. In three cases, Urdu nouns are inflected for two numbers, i.e. singular and plural, direct, oblique, and vocative. Urdu verbs have persons, numbers, gender, tense, mood, and aspect categories. The Urdu verb form agrees in gender and number with the subject.

B. Inflection

The affixations that precede or follow a language's root depend on the language typology [6]. The inflections of the language itself decide the concatenative and non-concatenative morphology. Many languages combine between regularity and irregularity of their word formation. The decision to address the morphology of a language as concatenative or non-concatenative depends on the grammarians. It is not valid for a language to be called non-concatenative language if some irregular word-formation processes are detected. What criteria have been followed to cluster language inflections or derivational? Like other languages, Urdu has regular and irregular word formations [7]. The distinction of gender and case in nouns in Urdu is inflected for two classes: the first is the number, singular or plural, and the second is based on declension, marked and unmarked types. Table I shows inflection paradigms in nouns. Table II shows noun inflections with examples [8]. The distinction is overtly noticed in number, gender, and case.

TABLE I. INFLECTION PARADIGMS IN NOUNS

		Singular		**Plural**		
		Direct	**Oblique**	**Direct**	**Oblique**	**Vocative**
Masculine	1st	*-ã*	*-e*		*-õ*	*-o*
	2nd				*+õ*	*+o*
Feminine	1st	*-ī, -i, -I, -yã*		*-iyɛ̃*	*-iyõ*	*-iyo*
	2nd			*+ɛ̃*	*+õ*	*+o*

TABLE II. NOUN INFLECTION EXAMPLES

		Singular		**Plural**		
		Direct	**Oblique**	**Direct**	**Oblique**	**Vocative**
Masculine	1st	lark*ã* ku*ɛ̃*	lark*e* *kuɛ̃*		lark*õ* *kuõ*	lark*o*
	2nd	*seb* *vãlid* *cãkū* *ādmĩ*			sebõ vãlidõ cãkuõ ādmiyõ[4]	*pitão* *ādmiyo*

II. URDU COMPUTATIONAL MORPHOLOGY

Urdu computational morphology studies the way Urdu forms and structures words [9]. This field is concerned with analyzing and understanding the rules and patterns that govern the formation of words in Urdu and using this knowledge to develop computational tools to process and analyze Urdu text. One of the main challenges in Urdu computational morphology is dealing with the complex inflectional and derivational morphological processes present in the language [10-12]. For example, Urdu verbs have a rich system of inflectional forms, which can change depending on factors such as tense, aspect, and person. Urdu nouns can also be inflected to indicate case and number and can be formed through various derivational processes. To address these challenges, researchers in the field of Urdu computational morphology have developed several different tools and techniques. One popular approach is to use rule-based methods, where a set of rules are defined for each morphological process, and these rules are then applied to words to analyze and generate new forms. Another approach is the utilization of statistical methods, where patterns in large text corpora are used to train computational models that can automatically identify and analyze morphological forms.

In recent years, there has been a growing interest in using deep learning approaches for performing different Natural Language Processing (NLP) tasks, e.g. Text Classification (TC), Question Answering (QA), Name Entity Recognition (NER), Sentiment Analysis (SA), etc. [13-15]. In the same way, deep learning techniques have been used for Urdu computational morphology. These techniques are effective in learning complex patterns in the language and have been used to develop many state-of-the-art tools for morphological analysis and generation in Urdu [16].

Urdu is a morphologically rich language with a complex system of inflections and derivations [17-19]. The main inflectional categories in Urdu include gender, number, case, and tense, while the main derivational processes include affixation, compounding, and conversion. In computational morphology, the task of analyzing and generating the inflected and derived forms of words is known as morphological analysis and generation [20]. Some computational approaches have been proposed for Urdu morphological analysis and generation, including rule-based, finite-state, and machine learning-based methods. Rule-based methods rely on manually crafted rules to analyze and generate inflected and derived forms, while finite-state methods use finite-state transducers to model the morphological processes of a language [21, 22]. On the other hand, machine learning-based methods use data-driven techniques such as neural networks to learn the morphological patterns of a language. One of the significant challenges in the computational morphology for Urdu is the large number of inflected and derived forms that a word can have. The rich inflectional system of Urdu, combined with relatively free word order, makes it challenging to analyze and generate inflected and derived forms fully automatedly. However, recent advances in machine learning techniques have shown promise in addressing this challenge. The research on Urdu computational morphology is ongoing, focusing on developing more accurate and efficient morphological analysis and generation methods. This includes the use of neural networks, LSTM, and attention-based architectures for morphological analysis and generation, which perform well in Urdu.

III. RESEARCH QUESTIONS

The research questions for this study focus on the lexical processing of Urdu nouns inflection.

- Question one: Does the Urdu language have a concatenative inflective morphology system?
- Question two: Can finite-state machines process Urdu nouns systematically?

IV. RELATED WORK

Urdu and Hindi are quite similar on a spoken level, but Hindi is written in Devanagari script, which is entirely different from the Urdu script. Regarding linguistics, morphology refers to the formation of words by focusing on their internal structure. Morphology is divided into two classes: inflectional morphology and derivational morphology. NLP is the primary interface between humans and computers. It is considered a branch of Artificial Intelligence (AI). Authors in [23] studied rule-based stemmer in Urdu and presented a rule-based stemmer for Urdu. They stated that stemming is a process in which affixes are separated from their root word. Their work was restricted to the Urdu language, which meant that the root word was not extracted from other languages like English, Persian, and Arabic. This stemming algorithm separated prefixes, suffixes, and infixes. It matched the word with the affixes and pulled out the word. The system and evaluation exhibited 86.5% accuracy. Authors in [24] developed an approach for building a finite-state morphological analyzer for Urdu and Hindi that is compatible with the XFST tools already used in both scripts. That was a part of the ParGram project, and they developed a grammar for South Asian Urdu. Very few resources exist for this language, and no broad-coverage finite-state morphological analyzer existed before this study. The approach allowed for an underlyingly similar treatment of Urdu and Hindi via a cascade of finite-state transducers that transliterate the very different scripts into a standard ASCII transcription system. The paper has addressed several issues in building a finite-state morphological analyzer for Urdu. The authors further explored the reduplication in Urdu, again based on solutions proposed concerning XFST. The discussion in their study mainly revolves around where and how information should be encoded concerning the morphology-syntax interface that has been defined between finite-state morphological analyzers and lexical-functional grammar. An improved statistical morph analyzer for Hindi, Urdu, Telugu and Tamil, was conducted in [25]. The authors proposed a statistical morph analyzer for Indian languages. The gender, number, person, and case accuracy was 86.81% for Telugu and 78.97% for Tamil. They studied two families of Indian languages, Indic and Dravidian, because most Indian languages fall into these two groups. Morphological analysis for Indian languages was conducted using the rule-based approach. Experiments were conducted for the four Indian Languages [ILs] mentioned above. The results were compared to Morfette (M) [26] and SMA [27] .The authors reported that for all four ILs, SMA++ outperforms the other SMAs. For Hindi and Urdu the L+G+N+P+C accuracy was 85.87%. For Telugu, the accuracy was 86.81%, and for Tamil, it was 78.97%. Authors in [28] reported a rule-based implementation of a morphological analyzer for Hindi. The study focused on designing a morphological analyzer for Hindi, a language rich in morphemes, which can be used for machine translation, which works on a rule-based approach and maintains a database for exceptions. The authors propose that the system's accuracy is very high as all the possible exceptions are covered. The word sense disambiguation can be integrated with this analyzer so that the words having multiple senses can be analyzed accurately. Authors in [29] studied the influence of morphology on word recognition in Hindi and Urdu. Hindi showed priming for morphologically and form-related primes, but the morphology study was not advantageous. Urdu showed faster latencies for targets preceded by morphologically related primes relative to those preceded by form-related primes. The evidence shows that the phonological, orthographic, and semantic/syntactic characteristics influence the visual recognition of the words presented in isolation. Targets were Urdu words of one syllable paired with two-syllable primes of three types.

V. FINITE STATE MACHINE (FSM)

An FSM is a mathematical model representing a language's morphological processes [30]. In the case of Urdu, an FSM could be used to model the language's inflectional and derivational processes and generate inflected and derived forms of words. FSMs are used to identify the regularity forms in a language by using a regular expression to perform the process of inflection morphology. The computational and linguistic components are represented together to produce the correct grammar forms using FSMs. The machines are used in NLP to construct lexical analyzers and to compile derivational and inflectional rules of language morphology using finite state tools, and a lexical analyzer was developed for the Urdu nominal system. A regular expression and the inflectional rules of the natural language can represent the set of strings S. The processing and the output can be represented and recognized by the FSA as shown in Figure 1.

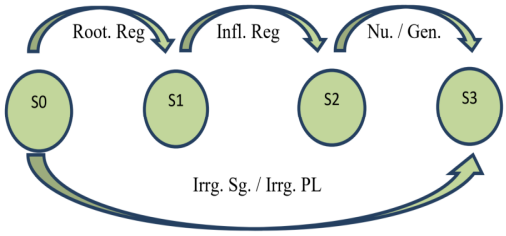

Fig. 1. Finite-state automaton.

Formally, a finite-state is represented by M =(S, ∑, s0, δ, F) where: S a finite set of states, S= {S0, S1, S2, S3}, Σ is a finite set of input symbols called the alphabet (root. reg, Infl. reg, Num., Gen), δ is a transition function (δ: S × Σ → S), S_0 is the start state ($s_0 \in$ S), and F is a set of finite states called accepting or final states. In Figure, it is F ⊆ S3).

VI. LEXICON

Lexicon generates entries for computational morphological analysis in the framework of finite-state morphology and uses

the concepts of the Xerox/Helsinki Finite-State Transducer LEXC Tools. Dictionaries often list many inflection classes and different types of LEXC entries [4]. A lexicon entry needs a general method that is prepared to accept almost any surface word. The resulting entries are combined with affixes and composed-intersected with the rule FSTs. The string pairs represented by the FST are produced in a human-readable form. The system is extensively redesigned and built using Xerox finite-state technology. Figure 2 shows a lexicon sample for creating the rules and the necessary morphological information for Urdu noun forms. The symbols used in the lexicon are based on the root and the inflectional information and are added based on rules which process all the entries and produce the output.

```
|----+----1----+----2----+----3----+----4----+----5----
 1   Multichar_Symbols +PL stands for plural
 2   +SG stands for singular +MS stands for masculine
 3   +F stands for Feminine +D stands for direct
 4   +O stands for oblique +V stands for vocative
 5
 6 LEXICON  Root
 7          Nouns;
 8
 9 LEXICON  Nouns
10 lark  Nenda;
11 bacchi Nendyan;
12 dawayi Nendyan;
13 kalam  Nendein;
14 rasta  Nende;
15 pankha Nende;
16 kamra Nende;
17 tasveer Nendei;
18 jangal Nendaat;
19 vālid  Nendō;
20
21 LEXICON  Nenda
22 +N+MS+SG+D:a #;
23 +N+R:0 #;
24
25 LEXICON  Nendyan
26 +N+PL:yan #;
27 +N+SG:0 #;
28
29 LEXICON  Nendein
30 +N+PL:ein #;
31 +N+SG:0 #;
32
33 LEXICON  Nende
34 +N+PL:e #;
35 +N+SG:0 #;
36
37 LEXICON  Nendei
38 +N+PL:ei #;
39 +N+SG:0 #;
40
41 LEXICON  Nendaat
42 +N+PL:aat #;
```

Fig. 2. Lexicon sample.

VII. RESULTS

After performing the morphological parsing and building the dictionary using the finite-state algorithm, given the roots and linguistic information, a determination algorithm is used to get the p-subsequential transducer. The determinization algorithm performs the composition in which the transitions are labeled with the input and output linguistic features. They must be added to the root as the input. The output is the various word forms along with linguistic features. Figure 3 represents the process of morphological parsing for the noun forms of Urdu. The processing starts from zero state to state one, where the base form is linked with all the possible forms, such as plural, number, gender, and case. For example, in the roots */lark/, and /ādm/*, the transitions are labeled with the category and inflection features, such as noun, masculine/feminine, and singular/plural suffixes. Another example is processing the root with the oblique/vocative forms and all other possible forms. The components seen in Figure 3 are:

- Thirteen states
- The first state is s0 and the final state is s12.
- 21 transitions (arcs)
- Σ: {(roots: lark, adam), (suffixes: *ā, õ, ī, iyã̃, iyo, e, iyõ)}*
- Used symbols: +PL/P refers to the plural form, +SG/S stands for singular, +MS/M stands for masculine, +F stands for feminine, +D stands for direct, +O stands for oblique, and +V stands for vocative.

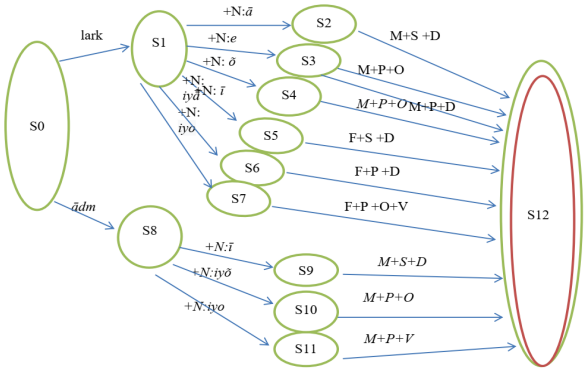

Fig. 3. Data representation sample using finite-state automata.

VIII. TRANSITION FUNCTION MATRIX

Table III shows the transition process of the FSM and the linguistic layers of morphological forms of the proposed model.

TABLE III. TRANSITION TABLE

From	To	Output
0	s1, s8	lemmas
s1, s8	s2, s3, s4, s5, s6, s7, s9, s10, s11	Various inflections
s1, s2, s3, s4, s5, s6, s7, s9, s10, s11	s12	All the possible linguistic features

```
xfst[7]: read lexc Mahyoob_Hindustani.txt
Reading from 'Mahyoob_Hindustani.txt'
Root...1, Nouns...10, Nenda...2, Nendyan...2, Nendein...2, Nende...2,
Nendei...2, Nendaat...2, Nend|...2
Building lexicon...Minimizing...Done!
3.3 Kb. 70 states, 85 arcs, 20 paths.
Closing 'Mahyoob_Hindustani.txt'
xfst[8]: print words
lark<+:a><N:0><+MS:0><+SG:0><+D:0>
lark<+:0><N:0><+:0><R:0>
bacchi<+:y><N:a><+PL:n>
bacchi<+:0><N:0><+SG:0>
dawayi<+:y><N:a><+PL:n>
dawayi<+:0><N:0><+SG:0>
kalam<+:e><N:i><+PL:n>
kalam<+:0><N:0><+SG:0>
kamra<+:e><N:0><+PL:0>
kamra<+:0><N:0><+SG:0>
rasta<+:e><N:0><+PL:0>
rasta<+:0><N:0><+SG:0>
pankha<+:e><N:0><+PL:0>
pankha<+:0><N:0><+SG:0>
tasveer<+:e><N:i><+PL:0>
tasveer<+:0><N:0><+SG:0>
jangal<+:a><N:a><+PL:t>
jangal<+:0><N:0><+SG:0>
vnlid<+:|><N:0><+MS:0><+PL:0><+O:0>
vnlid<+:0><N:0><+SG:0>
xfst[8]:
```

Fig. 4. Data processing with linguistic information.

Figure 4 represents the transition processing of the deterministic algorithm using the developed lexicon. For

instance, this sample has 10 roots and 12 inflection forms. The transition performs using 70 states, 85 arcs, and 20 paths. All the possible forms are generated with other linguistic information related to nouns in Urdu. Figure 5 shows a sample of one possible output wordlist of an FSM for the Urdu language and the readable output with all the needed features of the Urdu nominal systems. The first generated form is the singular and the plural nouns. Then the number, gender, and person singular and plural nouns are generated.

The output data extracted from the developed lexicon are evaluated with the correct forms of the different types of nouns in Urdu: the proposed computational algorithm and linguistic analysis were assessed with the high accuracy of 92.7%. Table IV displays the obtained results.

```
    |----+----1----+----2----+--
 1  l a r k + N +MS +SG +D
 2  l a r k a 0  0   0  0
 3
 4  l a r k + N + R
 5  l a r k 0 0 0 0
 6
 7  b a c c h i + N +PL
 8  b a c c h i y a  n
 9
10  b a c c h i + N +SG
11  b a c c h i 0 0  0
12
13  d a w a y i + N +PL
14  d a w a y i y a  n
15
16  d a w a y i + N +SG
17  d a w a y i 0 0  0
18
19  k a l a m + N +PL
20  k a l a m e i  n
21
22  k a l a m + N +SG
23  k a l a m 0 0  0
24
25  k a m r a + N +PL
26  k a m r a e 0  0
27
28  k a m r a + N +SG
29  k a m r a 0 0  0
30
31  r a s t a + N +PL
32  r a s t a e 0  0
33
34  r a s t a + N +SG
35  r a s t a 0 0  0
36
37  p a n k h a + N +PL
38  p a n k h a e 0  0
39
40  p a n k h a + N +SG
41  p a n k h a 0 0  0
```

Fig. 5. Singular and plural output wordlist sample.

TABLE IV. STATISTICAL ANALYSIS DATA AND RESULTS

Data (roots)	Correct output	Generated forms	Accuracy
2000	95.4%	30.000	92.7%
1000	96.6%	15.000	90.4%

The above statistical analysis shows that Urdu has a concatenative morphological system. The Deterministic Finite-State Morphological Analyzer tool is designed to analyze the morphology of Urdu nouns. It operates based on a FSM, which allows it to accurately identify and classify the various morphemes that make up a given word. This tool is highly effective in identifying the root of a word, as well as its various inflections and derivations. Its deterministic nature ensures consistent and reliable results, making it an essential tool for researchers working with language data.

IX. DISCUSSION

In this work, an attempt was made to develop a morphological analyzer for the low-source language Urdu. The first research question focused on the concatenative nature of Urdu noun forms. The study utilized an FSM to generate a morphological analysis of Urdu nouns. Figure 3 displays the regularity in most of the nominal forms of Urdu, which is consistent with previous research on the topic [9, 18, 24]. The study's findings reveal that Urdu has a concatenative system in the context of the noun forms, consistent with [18] in which it was reported that Urdu has a concatenative inflective morphology system. However, there are some exceptions when it comes to causative verbs, which can exhibit stem-internal changes in some instances. The data highlight the significance of comprehending the linguistic characteristics of all potential inputs that can be utilized by the tool. This understanding can ultimately result in improved output quality.

The finite-state tool was utilized to process noun inflection in Urdu. Figure 2 shows the lexicon development and rule generation to enable the FSM to produce all possible noun forms, each with its corresponding linguistic description. This approach effectively addressed the research question two. This study analyzed the rules and patterns that govern the formation of words in Urdu. The goal was to develop computational tools to process and analyze Urdu inflections. By understanding the intricacies of Urdu word formation, the accuracy and efficiency of language processing tools can be improved. The output displays various inflected forms of Urdu nouns, such as singular and plural forms, cases, and different grammatical genders. The specific morphological processes provided by the finite state tools and the detailed design of the FSM determine the actual output. The accuracy score of the analysis is above 90%. The FSM ensures accuracy and completeness in analyzing and creating these forms. This approach achieves a comprehensive understanding of noun forms, allowing for precise communication in written language.

X. CONCLUSION

This paper focused on the Urdu computational morphology using finite states to generate a morphological analysis for Urdu nouns. This study is concerned with analyzing and understanding the rules and patterns governing the formation of Urdu words. The objective is to use this knowledge to develop computational tools to process and analyze Urdu nouns. The output showcases Urdu nouns' inflected and derived forms, including singular and plural forms, various cases, and grammatical genders. The specific morphological processes used by the finite state tools, along with the detailed design of the finite state machine, determine the precise output. Overall, this paper contributes to computational morphology by comprehensively understanding finite-state technology's use in analyzing Urdu nouns' morphology. This study is limited to

Urdu noun forms analysis, and future studies will focus on other lexical categories of Urdu.

REFERENCES

[1] M. G. A. Malik, C. Boitet, and P. Bhattacharyya, "Hindi Urdu machine transliteration using finite-state transducers," in *22nd International Conference on Computational Linguistics*, Stroudsburg,PA, USA, Aug. 2008, pp. 537–544.

[2] R. Ahmad, "Urdu in Devanagari: Shifting orthographic practices and Muslim identity in Delhi," *Language in Society*, vol. 40, no. 3, pp. 259–284, Jun. 2011, https://doi.org/10.1017/S0047404511000182.

[3] K. V. S. Prasad and S. M. Virk, "Computational evidence that Hindi and Urdu share a grammar but not the lexicon," in *3rd Workshop on South and Southeast Asian Natural Language Processing*, Mumbai, India, Dec. 2012, pp. 1–14.

[4] K. Koskenniemi, "Guessing lexicon entries using finite-state methods," in *4th International Workshop for Computational Linguistics for Uralic Languages*, Helsinki, Finland, Jan. 2018, pp. 59–75, https://doi.org/10.18653/v1/W18-0206.

[5] J. M. M. A. Algaraady, "Needs Challenges and Preliminary Solutions for Verb Phrases Translation from English to Arabic An Example Based Machine Translation Model," Ph.D. dissertation, Aligarh Muslim University, Aligarh, India.

[6] K. V. Goethem, "Affixation in Morphology," in *Oxford Research Encyclopedia of Linguistics*, Oxford, England: Oxford University Press, 2020, pp. 1–35.

[7] S. Vikram, "Morphology: Indian Languages and European Languages," *International Journal of Scientific and Research Publications*, vol. 3, no. 6, pp. 1–5, 2013.

[8] M. C. Shapiro, "Chapter Seven: Hindi," in *The Indo-Aryan Languages 2*, 2003, pp. 276–314.

[9] A. Niazi, "Morphological Analysis of Urdu Verbs," in *17th International Conference on Intelligent Text Processing and Computational Linguistics*, Konya, Turkey, Apr. 2016, pp. 284–293, https://doi.org/10.1007/978-3-319-75477-2_19.

[10] T. Fatima, R. U. Islam, M. W. Anwar, M. H. Jamal, M. T. Chaudhry, and Z. Gillani, "STEMUR: An Automated Word Conflation Algorithm for the Urdu Language," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 21, no. 2, Aug. 2021, Art. no. 35, https://doi.org/10.1145/3476226.

[11] K. Riaz, "Challenges in Urdu stemming: a progress report," in *BCS IRSG Symposium: Future Directions in Information Access*, Swindon, United Kingdom, Aug. 2007, pp. 23–27.

[12] D. Chopra, N. Joshi, and I. Mathur, "A Review on Machine Translation in Indian Languages," *Engineering, Technology & Applied Science Research*, vol. 8, no. 5, pp. 3475–3478, Oct. 2018, https://doi.org/10.48084/etasr.2288.

[13] M. Mahyoob, J. Algaraady, M. Alrahiali, and A. Alblwi, "Sentiment Analysis of Public Tweets Towards the Emergence of SARS-CoV-2 Omicron Variant: A Social Media Analytics Framework," *Engineering, Technology & Applied Science Research*, vol. 12, no. 3, pp. 8525–8531, Jun. 2022, https://doi.org/10.48084/etasr.4865.

[14] J. Algaraady, "An analysis of Yemenis' responses and sentiments on social media towards the emergence of the COVID-19 pandemic," *Humanities and Educational Sciences Journal*, vol. 27, pp. 589–607, Dec. 2022, https://doi.org/10.55074/hesj.v0i27.621.

[15] J. Algaraady and M. Mahyoob, "Public Sentiment Analysis in Social Media on the SARS-CoV-2 Vaccination Using VADER Lexicon Polarity," *Humanities and Educational Sciences Journal*, vol. 22, pp. 591–609, Apr. 2022, https://doi.org/10.55074/hesj.v0i22.476.

[16] S. Jha, A. Sudhakar, and A. K. Singh, "Multi Task Deep Morphological Analyzer: Context Aware Joint Morphological Tagging and Lemma Prediction." arXiv, Sep. 16, 2019, https://doi.org/10.48550/arXiv.1811.08619.

[17] P. Sharma and N. Joshi, "Knowledge-Based Method for Word Sense Disambiguation by Using Hindi WordNet," *Engineering, Technology & Applied Science Research*, vol. 9, no. 2, pp. 3985–3989, Apr. 2019, https://doi.org/10.48084/etasr.2596.

[18] M. Humayoun, H. Hammarstrom, and A. Ranta, "Urdu Morphology, Orthography and Lexicon Extraction." arXiv, Apr. 06, 2022, https://doi.org/10.48550/arXiv.2204.03071.

[19] S. Mukund, R. Srihari, and E. Peterson, "An Information-Extraction System for Urdu---A Resource-Poor Language," *ACM Transactions on Asian Language Information Processing*, vol. 9, no. 4, Sep. 2010, Art. no. 15, https://doi.org/10.1145/1838751.1838754.

[20] M. Mahyoob, "Semi-automatic Annotation of Arabic Corpus: A Morpho-syntactic Study," Ph.D. dissertation, Aligarh Muslim University, Aligarh, India, 2015.

[21] M. Mahyoob and J. Algaraady, "Towards Developing a Morphological Analyser for Arabic Noun Forms," *International Journal of Linguistics and Computational Applications*, vol. 5, no. 3, pp. 45–51, Jun. 2018, https://doi.org/10.30726/ijlca/v5.i3.2018.52012.

[22] M. Mahyoob, "Developing a Simplified Morphological Analyzer for Arabic Pronominal System," *International Journal on Natural Language Computing*, vol. 9, no. 2, pp. 9–19, Apr. 2020, https://doi.org/10.5121/ijnlc.2020.9202.

[23] V. Gupta, N. Joshi, and I. Mathur, "Rule based stemmer in Urdu," in *4th International Conference on Computer and Communication Technology*, Allahabad, India, Sep. 2013, pp. 129–132, https://doi.org/10.1109/ICCCT.2013.6749615.

[24] T. Bogel, M. Butt, A. Hautli, and S. Sulger, "Developing a finite-state morphological analyzer for Urdu and Hindi," in *Finite-state Methods and Natural Language Processing*, Potsdam, Germany: University of Potsdam, 2008, pp. 86–96.

[25] S. Srirampur, R. Chandibhamar, and R. Mamidi, "Statistical Morph Analyzer (SMA++) for Indian Languages," in *First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, Dublin, Ireland, Aug. 2014, pp. 103–109, https://doi.org/10.3115/v1/W14-5312.

[26] G. Chrupała, G. Dinu, and J. van Genabith, "Learning morphology with Morfette," in *Sixth International Conference on Language Resources and Evaluation*, Marrakech, Morocco, Dec. 2008, pp. 1–6.

[27] D. K. Malladi and P. Mannem, "Context Based Statistical Morphological Analyzer and its Effect on Hindi Dependency Parsing," in *Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*, Seattle, WA, USA, Oct. 2013, pp. 119–128.

[28] A. Agarwal, Pramila, S. P. Singh, A. Kumar, and H. Darbari, "Morphological Analyser for Hindi – A Rule Based Implementation," *International Journal of Advanced Computer Research*, vol. 4, no. 1, pp. 19–25, Mar. 2014.

[29] C. Rao, "Morphology in word recognition: Hindi and Urdu," Ph.D. dissertation, Texas A&M University, College Station, TX, USa, 2010.

[30] M. Hulden, "Finite-State Technology," in *The Oxford Handbook of Computational Linguistics*, R. Mitkov, Ed. Oxford, England: Oxford University Press, 2022, pp. 230–254.