IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 29 December 2023, accepted 23 February 2024, date of publication 8 March 2024, date of current version 19 March 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3373714

RESEARCH ARTICLE

Building a Multilevel Inflection Handling Stemmer to Improve Search Effectiveness for Urdu Language

ABDUL JABBAR[1], SAJID IQBAL[2,3], ABDULLAH ABDULRHMAN ALAULAMIE[2], AND MANZOOR ILAHI[1]

[1]Department of Computer Science, COMSATS University Islamabad, Main Campus, Islamabad 45550, Pakistan
[2]Department of Information Systems, College of Computer Science and Information Technology, King Faisal University, Hofuf 31982, Saudi Arabia
[3]Department of Computer Science, Bahauddin Zakariya University, Multan 60800, Pakistan

Corresponding author: Sajid Iqbal (siqbal@kfu.edu.sa)

This work was supported by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia (Grant No. 5992).

ABSTRACT Stemming is an essential step in various Natural Language Processing (NLP) applications and is used to reduce different variants of the query words to a standard form to avoid the vocabulary mismatch issue in Information Retrieval (IR) systems. Due to specific grammatical rules and complex morphological structures, finding an effective stemming algorithm in Urdu is a challenging task. Although, several stemming algorithms have been proposed for the Urdu text stemming; however, none of them extract the stem from multilevel inflected forms. In this context, according to the best of our knowledge, this is a first effort towards the proposition and evaluation of a novel Urdu Text Stemmer (UTS) that can deal with multi-level inflection forms in Urdu text. The experimental evaluation of the proposed scheme has been conducted on the text-based and word-based custom-developed corpus. The proposed stemming technique is rigorously evaluated and compared with state-of-the-art stemming algorithms. Experimental results demonstrate that UTS outperforms existing Urdu stemmers and achieves an accuracy of 94.92% and 91.8% on word corpus and text corpus, respectively. We also evaluated our proposed system in an Information Retrieval application for Urdu, using the Collection for Urdu Retrieval Evaluation (CURE) dataset. Our approach for information retrieval outperformed and improved both recall and precision metrics.

INDEX TERMS Stemmer, information retrieval, Urdu stemmer, lemmatizer, natural language processing, text mining.

I. INTRODUCTION

Text stemming is a complicated and crucial step in many query systems, indexing, web search engines, and IR systems [1], [2], [3], [4], [5], document classification [6], [7], and linguistic feature extraction [3]. It provides the benefit of reducing the storage requirements by truncating redundant terms [8]. It increases the matching possibility for comparing documents and unifying the vocabulary process. The stemming is a computational process that reduces all conflated words to the same root or stems by stripping derivational and inflectional affixes [9]. For example, English words like 'consisted', 'consistency', 'consistent', 'consistently', 'consisting', and 'consists' can be reduced to 'consist'.

In Urdu, there are two types of stemming algorithms which include affix stripping (rule-based) [10], [11], [12]. In Affix stripping methods, a set of rules is developed based on the morphological structure of the language. To extract a stem, specific affixes are truncated from one or both sides of the root/stem. Contrarily, statistical methods are used in statistical stemming to extract various features of the words. Examples of some adopted statistical methodologies are n-gram [13].

Urdu is a semantic language with a composite morphological structure. It is different from most of the Western languages [11]. The existing stemmers for the Urdu language commit over-stemming, under-stemming,

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

and miss-stemming errors [9]. Consequently, these errors decrease the efficiency of stemming algorithms. Whereas over-stemming occurs when two words of different classes stem to the same root, for instance, the Urdu word ہاتھوں [hathon/hands] and ہاتھی [hathi/elephant] are mistakenly merged, such as ہاتھ [hath/hand]. Furthermore, the under-stemming takes place when two words of the same group must not be stemmed to the same root, for instance, when the stemmer fails to conflate the words موتیں [Motein/ mortalities] and the word اموات [amwaat / mortality] to the common root form as موت [mout/mortal]. Mis-stemming is defined as taking off the affix that is an actual part of the word, for example, stemming the Urdu word بخار [bukhaar/fever] to خار [khaar/barb]. We notice that the mis-stemming error is frequently encountered in the Urdu language. The commonly used Urdu prefix ب [bay] and suffix ی [ye] may often be the actual letter of the word, for example, the Urdu words بقیا [baqya/the restand] and لڑکی [larki/girl].

The existing Urdu stemmers fail to capture the stem from multilevel inflection and derivation, as well as مہمل [Mohmil/ refer meaningless words] words. For instance, bigram words having co-suffix such as صوبےدار [soobay daar/officer of a province], a Mohmal word as a suffix like چوری چکاری [chori chaakari/stolen].

This paper proposes an Urdu Text Stemmer (UTS) to clip the multi-level inflections, derivations, and Mohmil words. The proposed algorithm consists of compound word reduction, truncating the prefixes, suffixes, co-suffixes, infixes, and Mohmil words. The uniqueness of this paper among existing works is summarized below:

1. Multi-level inflections are handled by UTS, while current Urdu stemmers do not consider it, for instance, با اخلاق [baikhlaq/Well-mannered] possess prefix با [ba] and some infixes letters, after striping, the affixes and derived stem is خلق [khulq/politness].
2. Co-suffixes are not handled in any present Urdu stemmer; however, UTS copes with them, for example, رشتے دار [rishte dar/relatives] is stemmed as رشتہ [rishta/relation].
3. Existing Urdu stemmers do not remove the Mohmil words; however, UTS deletes the Mohmil words and extracts the stem such as چوری چکاری [chori chaakari/stolen] is stemmed to چور [chor/thief].
4. Resources to develop a stemmer such as prefixes, suffix lists, and rules are provided in this article.
5. To the best of our knowledge, it is the first effort to handle multi-level inflections and derivations in the Urdu language. The experimental evaluations show that the proposed algorithm outperforms the competitor stemmers.
6. The performance of the algorithm is assessed through both direct and indirect evaluation techniques.

The rest of this paper is organized as follows: Section II provides a brief background of the Urdu language grammar and morphology. Section III describes the existing work carried out in the same direction. Whereas Section IV elaborates on the proposed technique and evaluation datasets. Section V describes the evaluation methods of stemmers. The experimental results are presented in section VI. The discussion and analysis of experiment results are mentioned in section VII. Finally, section VIII concludes the paper.

TABLE 1. Inflection examples in the urdu language.

Examples	Transform	Method
والد [waalid/father] → والدین [walidain/parents]	Dual to singular	suffixation
جیل [jail/prison] → جیل خانہ جات [jail khanah jaat/ prisons]	noun Transform	Multi-level suffixation
لڑکی [larki/girl] →لڑکیاں [larkian/girls]	Singular to Plural Transform	suffixation
مرض [marz /a disease] →امراض[amraaz /diseases]	Singular to Plural (Broken/irregular) Transform	infixation

II. BACKGROUNDS

Urdu is known as one of the Major languages of the world after English, with 527 million speakers around the globe [14]. The rapid increase in the quantity of Urdu web documents over recent years has created a dire need for improving the performance of IR and text classification systems. Therefore, developing an accurate stemmer is a crucial step for automatic Urdu language processing [5].

Urdu is a highly inflected language written from right to left in contrast to English. In contrast with the English language, Urdu uses a non-concatenated way to derive the morphemes, which are interwoven to form words. The position of affixes and stems is coupled by concatenating morphemes in the concatenated languages. Urdu morphemes are interwoven in such a way that it is hard to obtain the stem from pattern-less Urdu words. For example, the موتیں [motein / mortalities], موتوں [mouton/ mortalities] میت [maiyat /dead body], and اموات [amwaat / mortality] are derived from the root word موت [mout /death]. Hence, extracting the stem from this linear decomposition principle by state-of-the-art algorithms is challenging.

In the Urdu language, new words are coined by derivation and compounding [9]. In the derivation, affixes (prefixes and/or suffixes) are attached to the root word to coin a new word. Both prefixes and/or suffixes are concatenated to the root to modify the meaning. In Urdu, prefixes are added to the right of the stem, and suffixes are added to the left, such as (suffix) ی [chohti ye] + (root) اتفاق [ittafaq/ unity] +(prefix) نا [na] and become نااتفاقی [na-itefaqi/ Disunity].

In compounding, there are two completely independent and meaningful words or meaningful words, and an affix is joined together to make a compound word [18], [19]. For instance, خوش اخلاق [khush akhlaq/ well-mannered] in which both words are meaningful, but in another compound word like عبادت گاہ [ibadat gaah/ house of worship], standalone the (affix)گاہ[gaah] has no meaning. However, if such a meaning-

TABLE 2. Examples of loan prefixes in urdu.

Languages	Prefixes	Words
Persian	پیش[paish]	پیش خیمہ [paishkhaima/precursor]
Hindi	ان [un]	ان پڑھ [un parh/illiterate]
Arabic	لا [la]	لازوال [la zawaal/Everlasting]
English	سٹی [city]	سٹی ناظم [city naazim/city coordinator]

TABLE 3. Examples of loan suffixes in urdu.

Languages	Suffixes	Words
Persian	دانی [daani]	چائےدانی [chayedaani/Tea pot]
Hindi	ہٹ [hat]	چکناہٹ [chiknahat/oily]
Arabic	یت [yat]	شخصیت [shakhsiyat /personality]
English	سٹور [store]	کریانہ سٹور [karyana store/ grocery store]

less word is attached to some meaningful word, it can produce new meanings. The meanings of عبادت [ibadat/ worship] are changed when affixes are attached. A hybrid compound word is another form of compound words [17] in which two words of different languages are added to make a compound word such as (English word) ٹیکس [tax]+(Urdu word) غنڈا [ghunda / hooligan] become a compound word غنڈا ٹیکس [ghunda tax/ hooligan tax]. Reduplication [15], [16] is also a form of the compound word in which both words are slightly different from each other, such as روٹی ووٹی [roti woti/ bread] where ووٹی[woti] is a Mohmil word. In Urdu, grammatical changes occur through suffixation and infixation [18], [19]. The examples of suffixation and infixation in Urdu are given in Table 1. Multilevel inflection and derivations can also cause a change in the Parts of Speech (PoS) group.

Like English nouns, Urdu nouns are modified to signify possession, plurality, and agency. However, Urdu verbs are modified more expansively than English verbs. From a single Urdu verb, around 60 different forms can be generated [20]. Urdu is highly Persianized and Arabicized because it has been significantly influenced by Arabic and Persian in terms of vocabulary and sentence structure [8], [21]. Urdu has a few native affixes, most of which are borrowed from Persian and Arabic [22]. Examples of borrowed affixes are given in Table 2 and Table 3. According to Table 2 and Table 3, it is difficult for existing approaches to recognize the complicated stems. To address this issue for the Urdu language, that is truncation of Arabic and Persian affixes, we combined the template-based approach, affix striping, and reference lookup.

III. RELATED WORK

A wide range of stemming algorithms have been developed for various languages Including English [23], [24], [25], [26], [27], [28], [29], [30], Arabic [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], Kurdish [42], [43], Nepali languages [44], Sundanese language [45], Panjabi text [46], [47], Sindhi [48], Hausa language [49], Pashto [50], Manipuri text [51], Persian [52], and Urdu [10], [11], [12], [13], [53], [54], [55], [56], [57], [58], [59].

Many stemmers are designed for a few major languages like English and Arabic. A very effective affix-stripping approach based on the automatic purging of affixes from root words has not been considered in developing stemmers for the Urdu language [9]. In the subsequent text, some state-of-the-art stemmers are described and analyzed.

Alnaied et al. [38] combines morphology analysis and stemming to enhance Arabic IR. It defined the various stages of preprocessing, indexing, query processing, and evaluation, showcasing how these components work together to improve retrieval effectiveness in the context of the Arabic language. The paper presents the experimental results, comparing the performance of the proposed methodology against baseline methods or existing IR systems.

Alshalabi et al. [32] proposed a Broken Plural Rules (BPR) algorithm for stemming Arabic words, specifically irregular broken plural words. The BPR algorithm introduces several new rules for stemming irregular broken plural words. These rules are based on the morphological patterns of the Arabic language. The authors evaluated the effectiveness of the BPR algorithm on a standard Arabic dataset and found that it was able to extract the correct root of the word more accurately than existing algorithms. Bessou and Touahria [34] developed ESAIR (Enhanced Stemmer for Arabic Information Retrieval) using linguistic resources such as Arabic words dictionary to derive the stem of Arabic words. The proposed algorithm is based on a template-matching approach, which is a more sophisticated approach than traditional rule-based stemming algorithms. The proposed algorithm is evaluated on a standard Arabic dataset. The results show that the proposed algorithm outperforms traditional rule-based stemming algorithms in terms of accuracy and retrieval performance. Kaur and Buttar [46] combined the table lookup, and suffix identification and deletion approach to extract the stem from Punjabi verbs. The authors evaluated the proposed algorithm on a standard Punjabi dataset. They found that the algorithm was able to achieve a stemming accuracy of 95.21%. This is comparable to the accuracy of other stemming algorithms for Punjabi verbs. Alshalabi [37] developed a pattern-based method according to the word lengths 4 to 6 to identify the infixes. Prefixes and suffixes are also identified based on the length of the word by a predefined prefix and suffix list to extract the root words from the Arabic word. Alnaied et al. [38] design stemmer is called Arabic Morphology Information Retrieval (AMIR). It handles the

TABLE 4. The comparison of state-of-the-art stemmer with UTS.

Ref.	Main Idea	Limitation
Akram et al. [59]	This algorithm used prefixes and suffixes lists to remove the affixes (prefix/suffixes). Confusing affixes are handled by the exceptional lists of affixes.	• Infixes are ignored. • Multi-level suffixation was not handled. • Hybrid words are not handled. • Did not include the English loan affixes. • Arabic, Hindi, Persian, and English (prefix/suffix) were ignored. • The hybrid word did not treat. • Resources (Affixes rules) list not available openly. • Did not evaluate in Urdu IR system.
Khan et al. [61]	This system utilizes the predefined prefix and suffix list to derive the stem. To treat the infixes some patterns are defined.	• Multi-level suffixation was not handled. • The hybrid word did not treat. • Incomplete list of infixes handling rules. • Exception of infixes rules are also ignored. • Prefixes and suffixes are not provided for further research. • Did not test in Urdu NLP applications such as Urdu IR.
Jabbar et al. [11]	The proposed stemmer used a predefined affix list to derive the stem. To handle the infixes some rules are defined. Confusing affixes are handled by the table lookup method.	• Multi-level suffixation was not handled. • Hybrid words are not handled. • Did not include the English loan affixes. • A list of rules (prefixes and suffixes) is not given in the paper. • Arabic, Hindi, Persian, and English (prefix/suffix) were ignored. • Did test in Urdu IR system.
Fatima et al. [10]	STEMUR used the prefix and suffix list to remove the affixes. To handle the infixes some rules are defined.	• English borrowed words partially handled. • Confusing affixes did not handle. • But the exception of infixes is not handled. • Multi-level suffixation. • Did not include the English loan affixes. • The hybrid word did not treat. • List of prefixes and suffixes not mentioned in the paper. • Did not evaluate in Urdu IR system.
Jabbar et al. [UTS]	• The proposed system included the Hindi, Arabic, Persian, and English affixes to obtain the correct stem. Consequently, English loan words and hybrid words are stem efficiently. • Mohmil words are identified and removed. • A set of rules are designed to treat the infixes. • Confusing affixes are handled by the table lookup approaches. • All the lists of rules are provided in the paper for further research. • The complete list of prefixes and suffixes is provided in the paper for future research. • Performance is assessed by direct and indirect evaluation methods.	

derivation of morphemes with pattern matching approach and inflection morphemes removed by predefined prefixes and suffixes. AMIR preserves the morphological information of words, which is important for IR systems. The authors evaluate AMIR on a standard Arabic dataset and find that it outperforms traditional stemming algorithms in terms of retrieval performance. Alshalabi et al. [32] built the prefixes and suffixes list from letter one to five letters minimum word length is set to four letters and affixes (prefixes /suffixes) are deleted based on the word length. The evaluation of the proposed algorithm on a standard Arabic dataset shows the highest score of 68% of F-measure from their competitors' stemmers.

Mustafa and Rashid [43] proposed an improved rule-based stemmer for the Kurdish language. This stemmer tokenizes the query text after that normalization is performed in which an Arabic letter such as ي [yaa] is replaced with another Arabic letter ى [yeh]. By using a list of prefixes and suffixes Kurdish words are removed. Finally, the stop words are removed, and the stem words list is obtained. Saeed et al. [42] proposed an iterative rule-based stemmer that removes the longest affixes (suffixes and prefixes) from query words. Bolucu and Can [60] proposed a context-sensitive stemmer combined with POS for Agglutinative Languages.

The existing Urdu text stemmers are focusing on the challenge of processing the Urdu language morphologically. Akram et al. [59] developed the Assas-Band Urdu stemmer and defined the affix (prefix and/or suffix) and affix exception lists to remove the affixes and extract the stem from them. Khan et al. [53] presented an Urdu stemmer without affix exception lists and utilized a predefined affix list to remove the affixes. Husain et al. [13] used an n-gram approach to generate the suffixes that are chopped off using frequency-based and suffix-based length. Khan et al. [61] presented an Urdu stemmer using a template-based approach. It defined the templates to identify infixes. If a template matches with query word, then the corresponding rules are applied to extract the stem. Kansal et al. [62] developed an Urdu stemmer that produced a list of possible stems using appropriate affix rules. They presented a database of the stems with their frequency. The possible stem is searched in the database and high-frequency stems are derived. Gupta et al. [63] proposed the stemmer that initially checks the query word (in exception word list and stop word list). If the query word is not found in both lists, then true affixes are removed to derive the stem. Ali et al. [12] used the patterns to recognize and remove the infixes. They used them in Urdu short text classification. Jabbar et al. [11] proposed a MU stemmer that extracted the bigram compound words from the text and derived the stem.

Considering the above, the motivation in the present article is to devise a linguistic knowledge-based stemmer that handles English browed words, hybrid words, multilevel inflections, and derivations in the Urdu language. The comparison of the UTS and the state-of-the-art Urdu stemmers is presented in Table 4.

Algorithm 1 Stem Produce Function

```
read query text
String [] function stem_produce(string query_text)
// step 1
Preprocessing – Tokenize query_text where non-Urdu or stop words are removed and remaining words are added in SWFTL
// Step 2
For each token in SWFTL
   If a token is bi-gram or trigram
      apply CWR and update SWFTL at that index
   Else Keep the token unchanged, and go to step 3
End for
// Step 3
Tokenize SWFTL by hard space & add produced tokens to the OWL, and go to step 4
// Step 4
For each word in OWL
   If suffix removal and the recoding rule are matched
      apply the rule and add to FSL
   Else if the given word remains unchanged go to step 5.
      // Step 5
   Else If the affix removal rule applies to the word
      apply the rule and add the stem to FSL. go to step 6
   // step 6
      Else If the infix removal rule is applicable
   apply the rule and add the stem to FSL. go to step 7.
      // step 7
   Else If the word is found in the reference lookup table
      retrieve the corresponding stem and add it to FSL
   Else add the original word to the FSL
End for
```

IV. METHODOLOGY

In this section, the proposed Urdu Text Stemmer (UTS) is presented. The UTS is based on the removal of suffixes, co-suffixes, prefixes, infixes, and Mohmil words from the query words to find the correct stem as shown in Figure 1. UTS is based on seven main steps described in the following paragraphs.

The first three steps are preprocessing, reduction of compound words, and transforming the tokens (obtained in step 2) into unigrams. Steps 4 and 5 manipulate the unigram words and derive the stem by clipping the affixes, if any. Handling trigram words, and multi-level affixes including Mohmil words is a complex task. UTS extracts the content words from the query text. Then, it removes the affixes to obtain a stem. For query word, compound word reduction, suffix removal, and recoding, prefix and suffix removal, infixes matching and removal, and table lookup approaches are integrated and applied in a sequence to extract the stem.

Abbreviations used in the algorithm:

- SWFTL: Stop words free text list
- CWR: Compound word reduction rules
- OWL: one-word list

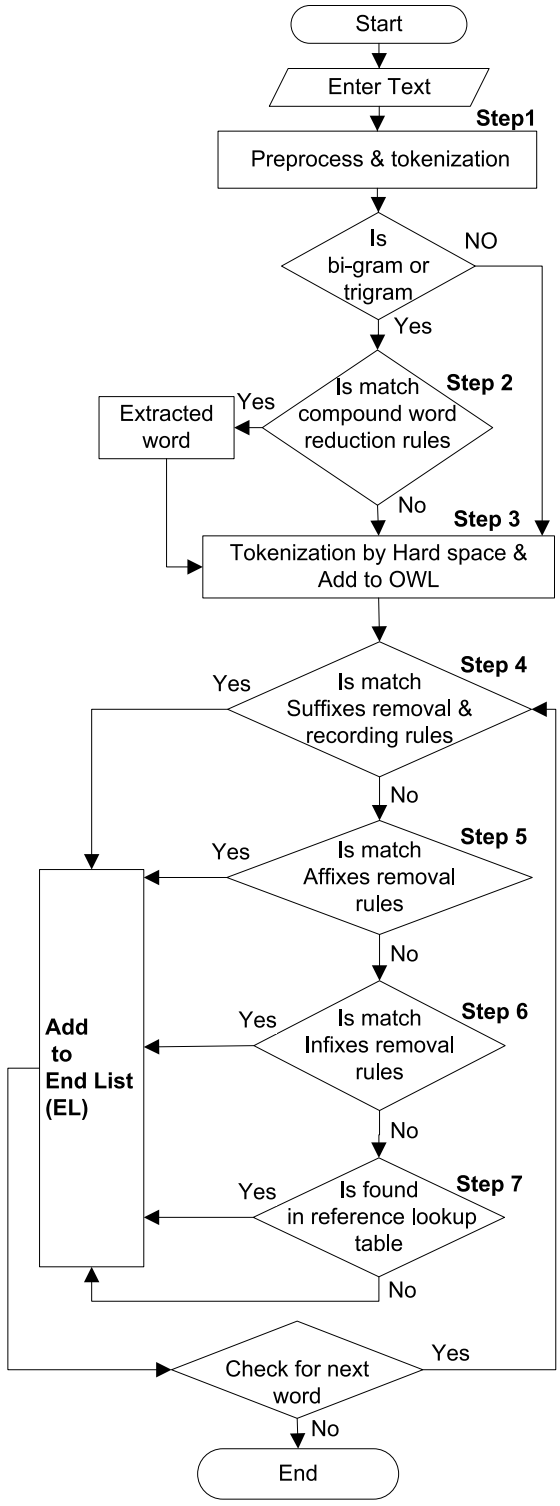

FIGURE 1. Proposed framework for UTS.

- FSL: Final Stem List
- SWL: Stem word list
- SL: Suffix List
- PL: Prefix list

The steps listed in Algorithm 1 provide an overview of the proposed methodology.

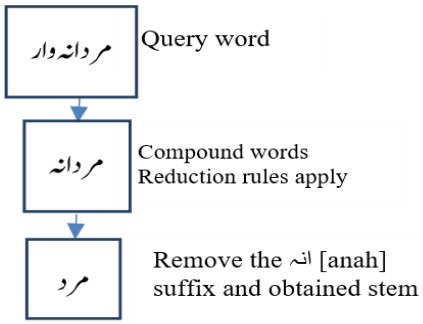

FIGURE 2. Example of the tokenization process.

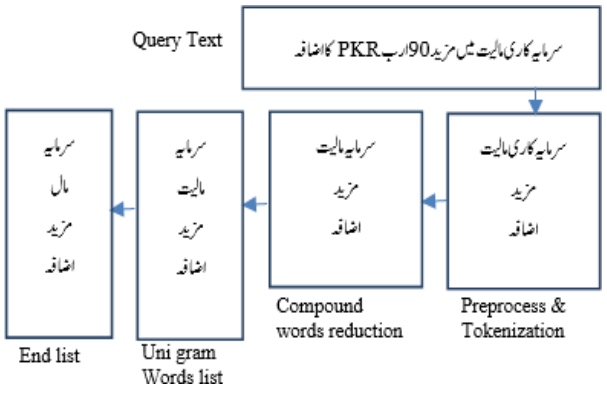

FIGURE 3. Example of compound word reduction rules.

TABLE 5. Mohmil affixes identification and removal.

Input Word	Criteria	Stem
پانی وانی [panivani/water]	The second word starts with و [wao] and the rest of the letters are alike in both words.	پانی [pani/water]
اکیلا دکیلا [akeladkela/alone]	The second word starts with د [dal] and the rest of the letters in both words are identical.	اکیلا [akela/alone]

Step 1: Preprocessing and tokenization

To create a vocabulary set, query text is tokenized. The tokenization marker list is mentioned in Appendix D. The tokenization is based on hard space and stops words (کا [ka/of], پر [par/on/at], سے [se/to]). An example of tokenization is mentioned in figure 2.

Step 2: Compound word reduction

The step consists of removing the compound affixes and Mohmil words. In the Urdu language, stem-able compound words (bi-grams and tri-grams) are formed by adding the affixes (prefix and/or suffix) or coined by adding Mohmil words as an affix with the root word. For example:

- The Urdu trigram words غیر تربیت یافتہ [ghairtar- biy-atYafta/ untrained] in which (Suffix) یافتہ [Yafta] + (root) تربیت [tarbiyat]+ (prefix) غیر [ghair], and derived stem is تربیت [train]
- The Urdu trigram compound word جیل خانہ جات [jail khanah jaat/ the prisons] that consists of (Suffix) جات[jaat] + (suffix) خانہ [khanah] + (root) جیل [jail], and the obtained stem is جیل [jail/ the prison]
- The Urdu bigram word با اخلاق [baikhlaq/ Well- mannered] possesses the prefix با [ba] and some infixes letters. After striping these affixes, the derived stem is خلق [khulq/politness].
- The Urdu bigram wor عقل مند [aqalmand/ wise] contains (Suffix) مند [mand] + (root ([aqal/ wisdom] and extracted stem is [aqal/wisdom]
- An Urdu compound word always contains the prefix ہمہ وقت [hama-waqt]. (Root) وقت [waqt/ time] + (prefix)ہمہ [hama], and the produced stem by the system is وقت [waqt/time].
- The Urdu compound word غلط سلط [ghalat salat/ wrong] in which (Mohmil words) سلط [salat] + (root) غلط [ghalat/ wrong], and غلط [ghalat/ wrong] is extracted as a stem.

The processed query word in this step may not be a final stem, for instance مردانہ وار [mardana waar/ by male], and produced word is مردانہ [mardana/ male] that is not a final stem as the word مردانہ [mardana/ male] still has انہ [ana] suffix, and to remove this, the extracted word (i.e., مردانہ [mardana/ male]) is passed to the next step. If compound word reduction rules do not match, then the unchanged query word passes to the next step. A sample of the compound word reduction system is shown in Figure 3.

Examples of rules of Mohmil word reduction are given in Table 5, and the complete list is given in Appendix B.

Step 3: Split the token into unigram

In this step, the token received by the previous step is split based on hard space, and a unigram words list is constructed. For example, if tokens obtained from the prior step (step 1) are بابانور سالہ [baba noor saalaah], then it is split based on hard space into two words بابانور [baba noor /name of a person] and سالہ [saalaah/ year].

Step 4: Suffix removal and recoding

The procedure takes a query word from the preceding step and checks the existence of the suffix based on the length of the word. If the suffix is found, then the conditions related to that rule are applied and the stem is derived. Otherwise, the original word is passed to the next step. When the number of rules attached to the criteria is more than one, then the derived stem is verified from the SWL (Stem Words List), for instance, a word ending in ے has three rules:

Rule 1: وعدے [waday/promises] وعد [wad] by removing the suffix ے [bri ye]

Rule 2: وعدے [waday/promises] وعدا [wada] by replacing suffix with [alif]

Rule 3: وعدے [waday/promises] وعدہ [wadah/promise] by replacing suffix with ہ [he]

In the above case, three possible stems are produced by the system using the above rules and each is checked in SWL and if found, is added to the EL (End List) as a stem. Otherwise, the original word is passed to step 5. Such as only وعدہ [wadah/promise] is found in SWL, which is added to the EL as a stem. The exceptions to these rules are handled by table

TABLE 6. Example of urdu suffixes.

lists	suffixes
List 1	و ،ا،ی
List 2	کش، یک ،سا
List 3	نما،جگر،غیر
List 4	خانہ،کاری،شدہ
List 5	انگیز ،ائیں،روائی
List 6	مزاحیہ،شایانہ،افزائی، خداداد
List 7	ترازوئے ،گزارانہ،مجرمانہ
List 8	سپرداری،جمالیاتی

TABLE 7. Examples of urdu prefixes.

lists	Prefixes
List 1	ب، ن
List 2	بن،کم،نا
List 3	ابن،غیر،ہمہ
List 4	ابدی،ابلق،اپی
List 5	ابتدائی،ابالی
List 6	گلوگیر،ابنائے
List 7	آرائیاں،آہنگیوں ،برادران
List 8	برادریوں،دستاویزی،مخالفانہ

lookup approaches such as کرائے [karaye/rental] is stemmed to کرایہ [kiraya/rent]. The complete list of suffix removal and recoding rules can be found in our research work [61].

Step 5: Circumfixes, prefix/suffix removal

The procedure initially checks the true circumfixes (both prefix and suffix) by predefined Prefix List (PL) and Suffix List (SL). If true circumfixes are found, they are removed to get the stem. For example: ناخوشگوار [nakhushgawaar/ Unpleasant] stems to خوش [khush/Happy]. If true circumfixes are not identified, then true prefixes are checked, if found, then the prefix is truncated, and the stem is added to the FSL. For instance, نوجوان [nojawan/younger] is stemmed to جوان [jawan/young]. If a true prefix is not found, then a check for a true suffix is performed and is removed if found and added to FSL; otherwise, the original word is passed to the next step. For example: زمیندار [zamindar/landlord] is stemmed to زمین [zamin/ land].

To avoid the under stemming and over stemming, affixes are removed according to the length of query words. The minimum query word length is set to four characters and the minimum produced stem length will be three characters if a two-character stem is derived then it is verified from the SWL. Here, we deal with a maximum of 8-character long affixes. These suffixes are arranged in descending order and removed with the longest match first. The example of suffixes is given in table 6 and prefixes are shown in table 7. A complete list of these affixes can be found in Appendix B and Appendix C.

Step 6: Infixes handling

This process identifies the infix letters using pre-defined patterns depicted in Table 8. In table 8 id contains a three-digit first digit that shows the length of the word and the remaining two are pattern id. The '-' dots in the pattern column can be replaced with any letters. As shown in Table 9 if infixes are found, then their corresponding rules are applied, and the stem is added to FSL. On the other hand, if no infixes are identified, then the original word is passed to the next step. When several rules are attached to a pattern, then the obtained stem is verified from the SWL, for instance, the pattern matched with the Urdu word ابدان [abdaan/ bodies] produces two stems:

TABLE 8. Patterns to identify the infixes.

Patterns	Id	Patterns	Id	Patterns	Id
ا---	401	ا---ء	501	ا-ا---	601
-ا--	402	ا---ا	502	ا---ء	602
--ا-	403	ا-ا-	503	و-ی-ن	603
-ی--	404	م-و-	504	ا-ت-ا-	604
-و--	405	ت-ی-	505	ا-ا-ی-	605
ا-ا	406	-ی--	506	-ا-ی-ن	606
ء--	407	-ء--	507	--ا-ی-ن	607
آ-ا	408	ا---ا	508	ا-ا-ا-	608
ا-ا-	409	ا-ا-	509		
م-و-	410	م-و-	510		
		ت-ی-	511		
		-ی--	512		

Rule 1. ابدان [abdaan/ bodies] بدن [bdan/body], remove first and fourth letter ا [alif]

Rule 2. ابدان [abdaan/ bodies] بدہ [bdah], remove the first and fourth letter ا [alif] and substitute ہ [he] at the end of the word.

The exception of these rules is handled by reference lookup table, for instance, احساس[ahsas/sensitives] where the corresponding stem is حس [ehs/a sense of] and اعداد [adaad/numbers], where عدد [adad/number] is the stem.

Step 7: References lookup

This step takes a query word and checks its existence in the table lookup, if found, then the corresponding stem is returned. Otherwise, the query word is included in the FSL. For instance, اساتذہ [asaatzaa/teachers] has its appropriate stem استاد [ustaad/ teacher].

V. EVALUATION CRITERIA

In the literature, several evaluation criteria have been suggested to evaluate the strength and accuracy of stemming algorithms [64], [65], [66]. Stemming evaluation methods are classified into two categories: direct and indirect evaluation The performance of the stemmer is directly evaluated using Sirsat et al. [64] evaluation metrics, and the precision, recall, and F-measure metrics. On the other hand indirect evaluation, the performance of the proposed system is evaluated in other applications such as information retrieval applications.

TABLE 9. Example of infix handling rules.

Pattern id	No. of Rules	Rules	Examples
509	2	Remove the first and fourth letter ا [alif]	احکام [ehkaam/orders] →حکم[hukum /order]
		Remove the first and the fourth letter ا [alif] and substitute ہ at the end of the word.	اتحاف [Ittehaf/gifts] →تحفہ [tohfa/gift]
604	1	Remove the first, second, and fifth letters from the query word.	اختتام [ekhtataam/ends] →ختم [khatam/end]

A. DIRECT EVALUATION METHOD

To directly compare the stemmers' performance without a specific application, we have chosen the Precision, recall, and F-measure and evaluation method of Sirsat [11], [61].

1) PRECISION, RECALL AND F-SCORE

In this section, we compared the produced stem by the stemmers with the human-extracted lemma list. The obtained results have been compared in terms of Accuracy (Eq. 1), Precision (Eq. 2), Recall (Eq. 3), and F-score (Eq. 4).

$$\text{Accuracy} = (TP + TN)/(TP + TN + FP + FN) \quad (1)$$

$$\text{Recall} = (TP)/(TP + FN) \quad (2)$$

$$\text{Precision} = (TP)/(TP + FP) \quad (3)$$

$$\text{F1(recall, precision)} = 2 * (Precision * Recall)/(Precision + Recall) \quad (4)$$

2) EVALUATION METHOD OF SIRSAT

This criterion is very compelling for assessing the strength and accuracy of a stemming algorithm. The following parameters are used to evaluate the strength and accuracy of the stemmer [64].

Index compression factor (ICF): ICF indicates the percentage by which a collection of distinct words is reduced by stemming. The ICF is defined in Eq. 5:

$$ICF = ns \quad (5)$$

where,

n = Total number of words before stemming

s = Number of words after stemming

Word Stemmed Factor (WSF): It is the average number of words that have been stemmed by a stemmer (as given in Eq.6). The threshold value is 50.

$$WSF = WS/TW * 100 \quad (6)$$

where,

WS indicates the number of stem words,

TW stands for the total number of words

Correctly Stemmed Word Factor (CSWF): The value of CSWF shows the accuracy of the stemmer as mentioned in Eq. 7. The minimum threshold is 50.

$$CSWF = (CSW/SW) * 100 \quad (7)$$

where,

CSW = correctly stem words, whereas

WS refers to the total number of stemmed words

Average Words Conflation Factor (AWCF): AWCF is the average number of variant words of different conflation classes that are stemmed correctly to the stem/root. To calculate AWCF, we first compute the number of unique words after conflation, which is calculated as follows Eq. 8:

$$NWC = SCW \quad (8)$$

where S shows the number of distinct stems after stemming,

CW refers to the number of incorrectly stemmed words. Then, AWCF is computed by Eq. 9:

$$AWCF = (CSW - NWC)/(CSW) * 100 \quad (9)$$

To evaluate the performance of the proposed stemming algorithm, a series of experiments is conducted.

B. INDIRECT EVALUATION METHOD

For indirect evaluation, we have chosen the BM-25 retrieval model based on the probabilistic retrieval framework [5]. The scoring function of BM25 takes into account various factors such as term frequency saturation, inverse document frequency smoothing, and document length normalization [67]. To assess the precision-enhancing ability of various stemmers, we analyzed their retrieval performance using recall@10, precision@10, and Mean Average Precision (MAP).

VI. EXPERIMENT SETUP

This section provides detailed descriptions of the dataset and presents the results obtained through both direct and indirect evaluation methods.

A. CORPUS DESCRIPTION

To evaluate UTS, we constructed the dataset which contains text fragments, including news articles (politics, literature, science, and technology) collected from BBC[1] Urdu and DAWN[2] news containing 20000 words, including stop words, verbs, adverbs, adjectives, nouns, proper nouns, punctuation marks, English words, numbers, and special symbols. Word corpus consists of 56074 Urdu words containing uni-gram, bi-gram, tri-gram compound words, broken plural words, and words with infixes. The dataset titled USED (Urdu Stemmer Evaluation Dataset) is collected from the following sources.

Four Urdu grammar books [16], [19], [68], [69].

1) Resources provided by [72] on Urdu morphology,
2) Online resources: Urdu online encyclopedia[3]

[1] http://www.bbc.com/urdu
[2] https://www.dawnnews.tv
[3] https://www.urduencyclopedia.com/

TABLE 10. Used dataset description.

Text corpus	Topic	No of articles	Total Words
1	Politics	5	6500
2	Literature	5	4600
3	Science	5	5300
4	Technology	5	3600
Total		20	20000

TABLE 11. Results of performance comparison.

Stemmers	Acc. (%)	Rec. (%)	Pre. (%)	F-score (%)
UTS (words)	94.92	99	95.58	97.32
UTS (Text)	91.8	97.48	93.88	95.65
Multi-Step [11] (words)	92.97	99	93.57	96.26
Multi-Step [11] (Text)	90.33	97.43	92.35	94.82
Khan et al. [61]	-	96.08	89.95	92.49
Assas Band [59]	91.2	-	-	-
Husain et al. [13]	84.27	-	-	-

3) CLE Urdu words list[4]
4) CLE Urdu high-frequency words list[5]

The word corpus has been preprocessed through the following steps:

- Elimination of Urdu diacritics.
- Removal of stop words, punctuation, numbers, and symbols.
- Deletion of English and French characters.

The text documents are related to 4 topics and each topic is represented by 5 texts with different lengths (Table 10), which lead to a total of 20000 words. The text corpus feeds the stemmer without preprocessing and tokens the text using the token marker mentioned in Appendix D.

The Urdu retrieval experiments have been carried out on 1096 documents from 254 domains with 50 queries from [70]

B. AFFIX STRIPPING EXPERIMENTS

To measure the performance of the proposed stemmer we involved human experts to annotate the words with the actual stem. They are native Urdu speakers with relevant qualifications. Obtained annotations are cross-validated by each other and results are used for the rule extraction which leads to the development of stemmer. The stemmer is applied to the raw data. The results produced are compared with human expert annotations. In this subsection, we compare the obtained results of UTS with existing Urdu stemmers Assas-Band stemmer [59], and MU stemmer [11]. The selection of stemmers is based on multiple factors. Assas-Band stemmer [59] stemmer has high accuracy among the rule-based Urdu stemmer and therefore is selected as representative of Urdu rule-based stemmers. Similarly, the MU stemmer [13] is better among infixes removal stemmers [12], [61]. Likewise, [59] is a statistical stemmer that is tested on Urdu text.

[4]http://www.cle.org.pk/software/ling_ resources/wordlist.htm
[5]http://cle.org.pk/software/ling_resources/ UrduHighFreqWords.htm

TABLE 12. Performance comparison using a standard dataset.

Evaluation metrics Sirsat [64]	Assas Band [59]	Multi-Step [11]	UTS
Total words (TW)	56074	56074	56074
No. of distinct words after stemming (S)	26597	25233	24970
Index Compression Factor (ICF)	53	55	**55.47**
No. of words stemmed	55128	54012	54179
Words Stem factor (WSF)	98.31	96.32	96.62
Correctly stem words (CSW)	49238	50693	51788
correctly stem words Factor (CSWF)	89.32	93.86	**95.59**
No. of distinct words after conflation (NWC)	24079	23795	23532
Average words conflation factor (AWCF)	51.10	53.06	**54.56**

Finally, Tables 11 and 12 show the performance measures of UTS compared with state-of-the-art Urdu stemmers. The majority of the existing Urdu stemmers (e.g. [10], [59], [61] are evaluated on the word corpus. Usal stemmer [55] also works on textual data. However, it uses hard space to identify the boundary of the words. In Usal stemmer, compound words are wrongly split into two unigram words; therefore, the compound word remains unstemmed. For instance, یہ عبادت گاہ ہے [yeh ibadat gaah hai/this is a place of worship] is tokenized form of the compound word عبادت گاہ [ibadat gaah/ place of worship] into two unigram word عبادت [ibadat] is a root word, گاہ [gaah] is a suffix, consequently compound word عبادت گاہ [ibadat gaah/ place of worship] remains unstemmed.

C. INFORMATION RETRIEVAL EXPERIMENTS

Table 13 compares the result of the retrieval performance of various stemmers in terms of Recall @10 (R@10), Precision@10 (P@10), and MAP. The bold values in the tables indicate the best performance. The percentage improvement relative to the baseline is mentioned in parentheses. UTS achieved the highest score of 0.5194 recall, 0.712 precision, and 0.2598 MAP. UTS improves the performance of retrieval by nearly 5% in average precision and 0.5 % MAP against unstemmed word retrieval.

VII. DISCUSSION

We conducted a comprehensive analysis of our UTS from three distinct perspectives in Section V. In order to accurately assess the performance of UTS, we have included a comparison with other leading stemmers currently in use, such

TABLE 13. Urdu information retrieval results.

Stemmer	R@10	P@10	MAP
No stem	0.4822	0.68	0.2585
Assas Band [59]	0.5067 (5.1)	0.704 (3.5)	0.2587 (0.1)
Multi-Step [11]	0.5088 (5.5)	0.704 (3.5)	0.2590 (0.2)
UTS	**0.5194 (7.7)**	**0.712 (4.71)**	**0.2598 (0.5)**

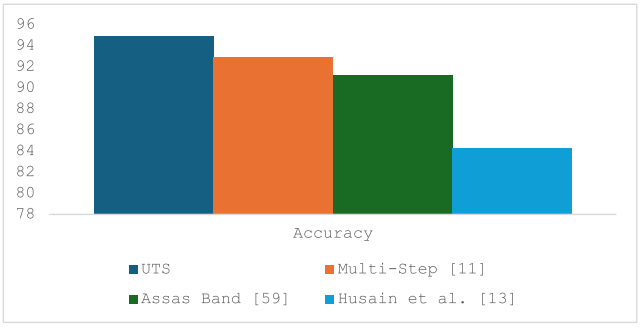

FIGURE 4. Comparison of results with respect to accuracy.

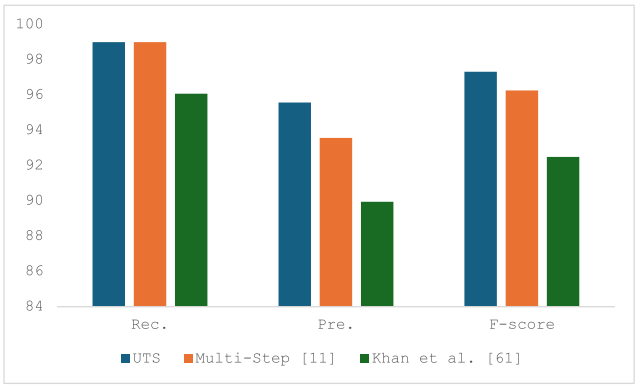

FIGURE 5. Comparison of results of precision, recall and F-score.

as multi-step [11], Khan et al. [61], Husain et al. [13] and Assas Band [59] stemmers. In addition, we also compare the complexity of the UTS algorithm.

We evaluated the strength and accuracy of UTS in our first experiment.

We performed two different experiments and compared in terms of accuracy and strength on the word corpus and text corpus. The experimental results are presented in Figure 4 to Figure 10. Assas Band stemmer [59] tested their stemmer on a corpus consisting of 21757 Urdu words and achieved 92.97% accuracy. Their stemmer removed prefixes and /or suffixes but did not handle the infixes. Husain et al. [13] proposed an N-gram stemmer to truncate suffixes and ignored the prefixes and infixes. This stemmer obtained 84.27% accuracy on a test size of 1200 Urdu words extracted from the E-mail corpus as depicted in figure 4.

Khan et al. [61] used a template to handle the infixes, but no mechanism is described to handle the exception of defined rules and template-less Urdu words. For instance, a pattern

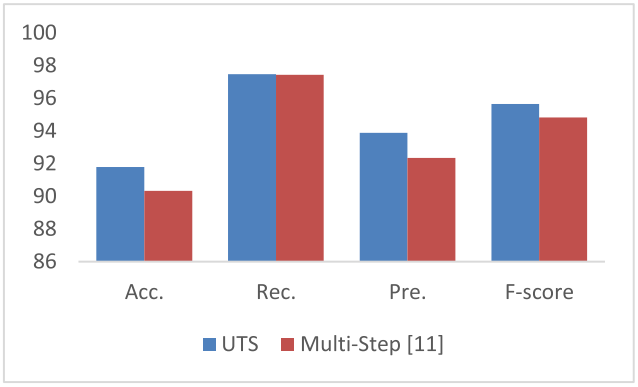

FIGURE 6. Comparison of results on text corpus.

is defined by Khan et al. [61], if an Urdu word has four characters and the third letter is و [vao], the third letter is removed to extract the stem, but this rule is violated in case of جلوس [juloos/ procession] and حصول [husool/ acquisition] to obtain the stem. This stemmer is evaluated on a corpus consisting of 19351 words and claimed precision and recall are 89.95% and 96.08, respectively as portrayed in Figure 5.

The MU stemmer [11] assessed both words and textual data and obtained a recall value of 99% and 95.586% precision. MU stemmer [11] extracted the bigram word from textual data to produce a stem, for example, the Urdu sentence یہ عبادت گاہ ہے [yeh ibadat gaah hai/this is a place of worship], after eliminating the یہ [yeh/this] and ہے [hai/is], the compound word عبادت گاہ [ibadat gaah/ place of worship] is extracted and the suffix گاہ [gaah] is removed to obtain the stem عبادت [ibadat]. However, their stemmer fails to deal with Mohmil words, and multilevel inflection and derivation. In comparison, UTS achieved recall and precision of 99% and 93.95%, respectively. On the text data set, the MU stemmer [11] yields 90.33% accuracy, followed by UTS which achieved an accuracy of 91.8%, as mentioned in Figure 6.

The results obtained on the same data set show that UTS achieved better performance than existing state-of-the-art MU stemmer [11] and Assas-Band stemmer [59]. Specifically, existing Urdu stemmers have caused some under-stemming errors for certain groups of words that hold multi-level inflection and derivation, for example: Bigram words having co-suffix تھانےدار [thaanaydaar/the officer of a police station], corresponds to the mistaken stem تھانے [police stations]. The Urdu bigram word تعلیم یافتہ [taleem Yafta/ educated], possessed suffix یافتہ [yafta] and some infixes letters, existing stemmer commit under stemming errors and produces تعلیم [taleem/education] as a stem. Bigram words having a Mohmil word as an affix سمجھابجھا [samjhabujha/understand] cannot be stemmed. Trigram words having prefixes and suffixes along with infixes, such as غیر تعلیم یافتہ [gher taleem yafta/uneducated], possess prefixes, suffixes, and infix. The existing Urdu stemmers produce an incorrect stem, i.e., تعلیم [taleem/education]. Trigram words having co-

TABLE 14. Sample output of the stemmers.

Query Words	Actual stem	UTS	AssasBand [59]	Multi-step [11]
چوری چکاری	چور	چور	چوری چکاری	چوری چکاری
نا تجربہ کار	تجربہ	تجربہ	تجربہ ک	تجربہ
امراض	مرض	مرض	امراض	مرض
بات چیت	بات	بات	ت چیت	بات چیت
مردانہ وار	مرد	مرد	مردانہ و	مردانہ
وجوہات	وجہ	وجہ	وجوہ	وجہ

suffix جیل خانہ جات [jail khanah jaat/ the prisons] and produce the incorrect stem جیل خانہ [jail khanah/ the prison].

On the text dataset, UTS also outperforms MU stemmers [11] (see Figure 7). The reason is, that MU stemmer [11] does not deduct the affixes from the token of three words size such as the obtained token سرمایہ کاری مالیت [sarmaya kaari maliyat/ worth of investment] consists of a compound word سرمایہ کاری [sarmaya kaari/investment], in which کاری [kaari] is an affix and سرمایہ [sarmaya/capital] is a stem. The compared stemmers with the proposed one do not remove the Mohmil affixes and multi-level affixes as shown in Table 14 and the incorrectly produced stems are underlined in the column. In Table 14, we can notice that all the words having Mohmil suffix چکاری [chaakari], چیت [cheet] are not stemmed. Whereas, in the case of multi-level affix مردانہ وار [mardana waar/manly] under stemming errors are committed by MU stemmer [11] and Assas-Band stemmer [43] UTS is the first Urdu stemmer that handles the multi-level inflections and Mohmil words reduction, as shown in Table 14.

The second experiment evaluated the strength and accuracy of Sirsat mechanisms for measurement. UTS obtained 95.59 % CSWF while MU stemmer [13] achieved 93.86 % and Assas-Band stemmer [59] obtained the lowest CSWF score of 89.32%. Assas-Band stemmer [59] blindly removed the affixes and achieved the highest WSF [98.31%] score than their competitor. The ICF achieved by UTS is 55.47 % value, which is higher than the competitor that has 55% value and 51 % as shown in table 12 and figure 7. Assas-Band stemmer [59] cannot handle the infix cases, so its performance is lower, as mentioned in Table 11, Table 12, and Figure 5. Whereas UTS extracts the correct stem and obtained CSWF is significantly higher which is 95.59% MU stemmer [11] obtained a score of 93.86 % and the Assas-Band stemmer [59] shows the lowest CSWF score of 89.32% as reflected in Table 11 and Figure 7.

The performance of the proposed UTS is comparatively higher for CSWF and AWCF scores as shown in Table 12. Therefore, from the obtained results using the Sirsats [64] evaluation method, we showed that the UTS provides better results in terms of performance, strength, and accuracy.

In the third experiment, we conducted a comprehensive analysis of UTS stemmers and their impact on Urdu information retrieval systems (indirect evaluation methods). The information retrieval results are mentioned in Figure 8-10. The UTS stemmer exhibits an enhanced MAP value compared to its competitors. It demonstrates a 0.5% improvement from the baseline, as illustrated in Figure 8.

TABLE 15. Mohmil (مہمل) words reductions criteria.

Input Word	Criteria	Stem Word
پانی وانی	Second word start wit و[wao] and rest of the letters are alike in both words.	پانی
اکیلا دکیلا	Second word start with و [wao] and rest of the letters are alike in both words.	اکیلا
چوری چکاری	First letters and last two letters match in both words	چوری
ٹھیک ٹھاک	First two letters and last letters match in both words	ٹھیک
حیص بیص	Second word start with ب[bay] and rest of the letters are alike in both words.	حیص
دھو دھا	First two letters match in both words	دھو
چاو چاوچوز	First one letter match in both words	چاو
خالی خولی	First one and last two letter match in both words	خالی
پیس پاس	First one and last one letter match in both words	پیس
سمجھا بجھا	Last three letters match in both words	سمجھ
پکڑ دھکڑ	Last two letters match	پکڑ
جھاڑو بہارو	Last one letter match in both words	جھاڑ
دھوم دھام	First two and last one letter match in both words	دھوم
پھینک پھانک	First and last two letters match in both words	پھینک
چوڑا چکلا	First and last one letters match in both words	چوڑا
بچا کھچا	Last two letters match in both words	بچا
بات چیت	Last one letters match in both words	بات

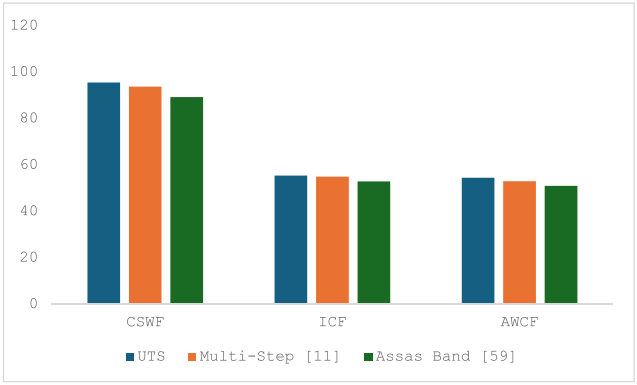

FIGURE 7. Results comparison on a standard dataset.

Figure 9 shows the better performance of UTS from their competitor stemmers. It attains the highest average precision and recall in the top 10 result as depicted in Figure 9.

The query-wise performance of UTS is presented in Figure 9. UTS achieves higher precision on a majority of queries, as seen in Figure 10.

TABLE 16. Collection of urdu prefixes.

List	Prefixes
1	م ب ن
2	ذو ہو لے مد گد کف کڑ کچ ظل طب صف ذی تل تپ پٹ جل مے تو دم شہ غم دل ہں نر کم نا زی بے با اک یک ہر نو لا سر تہ پر ہم آب اب ال آس شپ آل بر بڑ ھر ھم چو سر چی سش سہ ڈی پس بہ آؤ آن پچ پل پن تت تر تہ حق دل نم دو لب گل گز رگ رہ تر زر زن کٹ فن غل سگ شب شش شہ
3	ادھ سرد خار ہیڈ ماہ کڑی کچے کچی عطر عدت ضرب زیر زہر روح دشت برقی چوب جوش جام ٹکٹ تخت تال پیر پسر پاے پان جلا بخت ابر آہن آبی زود بال تنگ اشک نیک برگ بدھ باٹ باب اگن اگر اسم ابن غیر ہمہ ہفت ہشت زور نیم راہ آئر آبا ابو آبی ابی نور خوب ابل لوح لخت گلا گرم گرد کوے کوہ کور کند کشش آہن کرۂ کرہ ایک باج باد بار عون باگ بام عضو عرش عدم ببر بحث ظفر طبع ضعف بحر صلح صدر شوم شوخ شمع سیف سطح ستم سبک بدر ژرف زود برج زرہ رشک دلی دقت برق بڑا بزم داغ حکم حسن حسب حرف حبل بغل بلا بنی جنم جنت جمع جشن تنگ ثمر ثقل زبر تیغ تنگ تند تلا ترش تخم تحت تاو تاؤ تام گلو تار تاج عدل پیک نیل پون پنگ ناز پنج پکی پکا فور پست پدم قبل خرد پاس شپس جگر جگت تیز تیر تخم مسدس خود خوب تلخ
4	مدبر عددی طبعی صحیح شکست سبزہ زہرہ زنان دماغ حکمت حرکی حاصل چوبی جذبہ ترکی ترکش تحیت تازی تادم پنجہ پلنگ پستہ پتھر پائے پاؤں تخمی آبلہ سوڈو روئے ایکس انڈر آہنی باسی بازی بادہ بندہ برنج برتن آثار اپلا ابجد ابدی آبرو ابلا ابلغ ابلق اپیل مدار نظام خلاف ہوال دارل ناقص نازک مرکب ماہر ماتم لیزر لیڈی کورا مسلک مردم مذاق محفل مجلس مثبت مادر لباس گوشہ گوشہ گورا گندہ گلشن گرمی گرما گردش کورٹ کثیر کامل کاتب قومی آتشی فوجی عضوی پرست عالی طلاق الہی ضعیف انشا بابا باقی شوخی بانگ بایو بخشی شعلہ شعلہ سیرت سجدہ سجدۂ بدری سبزۂ بدعت سادہ برہم دامن زریں رویت روشن روبہ رونق رائے ذہنی بستر بسیط بعید حالت حاضر بقاے بگڑا بلند چہار چائے جوہر جذبہ ترنج جائے ثقلی تیغہ تیغہ تیزی تکیہ تکیہ تکلف تشنہ تشبہ ترنم صاحب بلیک بزرگ تراش تخمی تختہ تختہ برزخ تبسم تباہ تائے تائب باڈی تازہ تارک بادۂ تابہ تابع پیلی پہلا پھٹا گراں پنجہ پنبہ پنبہ بالا پارہ پختہ طالع پتلی پتخہ پبلک ابیض موجب پاکٹ پارہ پارۂ جلدی جلال جلاء تیرہ
5	مجازی متوسط گرمیء گردشی صحیفہ زنانہ جوابی ٹھنڈی تکبیر تشنہء تشبیہ تسخیر تحریف تحریر تائید تاریخ پچھلے ابالی ابجدی ابدان ابروے ابناء ابنای ابوال ابدال آتشیں اتمام آلودہ انگشت بازار باران بادہء آئینہ آوارہ بادیہ باراں باریک باطنی بدرجہ بندہء برہان برائے ولدال میثاق آخرال اخذال نورال ہومیو وقرال نوائے نظامت متبرک ماتمی لیڈیز کھوٹا کھوٹی مستقل مرکزی مذہبی مخملی مخلوط محکمہ محکمہ مجلسی متفقہ مادری ماتمی لشکری مابعد ٹرائی کشادہ قانون طر یقہ طالع مافوق صحیفہ شیریں شومیء شوخئی شوخیء شہادت سیاسی سلطان زنانی زاو یہ دماغی دعائے دارال خلافت حیثیت حاضری جوہری بدرجہ ثانوی جرائم ثلاثی ٹیڑھی تقصیر ٹھنڈا تیز یہ تیزاب تکمیل تکلیف تیرگی تقویم تقطیر جلالت تعویذ تعدیل تصفیہ تصفیہ پارٹی پارچہ پائین ترکیب تذکرہ تحلیل تحسین تحریم تحریک پچھلی پہاڑی شکستہ تجنیس
6	بہترین احقرال مائکرو ماہانہ مردانہ مجموعہ متوازی متنازع کورانہ طوفانی طبقاتی سلطانی سرکاری سالانہ زنجیری رومانی روحانی روایتی روحانی روایتی ڈسچارج خوشامد تیزابی تکمیلی تکبیری تقابلی تعدیلی تصدیقی تحو یلی پیمانہ ابنائے اتفاقی بازاری آئینہء پاکیزہ پبلشنگ پیمانہ تاریخی تاریک اولوال تبدیلی نافذال واحدال گلوگیر نادرال ناقابل ناجائز تدریجی ترجیحی مزاحیہ مخملیں مجموعی تقاضاے تقطیری مجموعہ تکوینی تولیدی متعلقہ گیسوئے ٹکسالی کاتبین قانونی عمرانی زاو یہء طر یقہء زہر یلی صوبائی شیرینی شاہانہ زہر یلے ژولیدہ سائنسی معروضی
7	قرارداد اظفارال کیمیائی کیمیاوی طلسماتی خاندانی ترازوئے ابتدائے ابتدائی سائنٹفک شیرینیء طلسمانی گورنمنٹ مجرمانہ محافظتی محکمانہ مدافعتی مشرکانہ مدافعتی نفسیاتی برادران صوبےدار
8	اصطلاحات دستاویزی مولو یانہ لیفٹیننٹ متصوفانہ کوہستانی جمالیاتی معصومانہ

The time complexity of the proposed algorithm is $O(n)$. Because the execution time is directly proportional to the size of the input. Steps 4 to 7 are executed in the nested loop which is based on the number of rules defined for the step. The fixed number of rules adds a constant factor in time complexity. Moreover, during asymptomatic analysis, lower-order terms and constants are ignored [59]. Similarly, space complexity also remains linear. At the start of the algorithm, data is loaded, which is then reduced in the following steps. Rule lists used to stem the words are of a fixed size, which adds a constant space complexity that can be ignored in space complexity analysis [59]. The time and space complexity of the UTS is better than the MU stemmer [11] which exhibits $O(n^2)$. The rest of the stemming methods have not discussed the time complexity and hence we are unable to compare them.

Although the better efficiency to produce stem has been achieved by UTS, however, there are some limitations which are faced by this algorithm such as it may mistakenly stem (False Positive) the proper noun, for instance, ارشاد [Irshad/Name of a person] is wrongly stemmed to رشد [rushad/ guidance]. The reason is that there is no mechanism in the Urdu language to identify the proper nouns. The Mohmil compound words that are not split by hard space such as کھاناوانا [khana wana/ the meal], patternless words, and confusing words that may be used as a root word or as

TABLE 17. Collection of urdu suffixes.

Lists	Suffixes
1	ت ا ی ہ ھ س ء ؤ ح ر و ز ٹ ڑ ں ۃ ک
2	ۂ تو ات ونی نک یت وا رس گی یں سٹ اؤ ہے وند نگ وں یز ہا سی یہ یا تا نہ شو رو جو بہ ہا بر ہے چہ چی را ا اں ہں ڑا ڑی ری زن ئی گر اک نا تر پن پا او ہٹ کش سا یو تی شا سہ تی تا دہ پن نے نا ئے ال ور ئی تا گو چی تے تی گر رہ اں یں ین کن رد چے وے اُوی لش ہی کن وں تہ نش نی ان نا
3	اؤں تیں ئیں جات واس باش ئیو بین توں اری یچہ یئن فس نوش روی توز بین کشی آرا کشی زدا نگر راں کدے انے ویز وتی اتی خور گین نہا ندی لیا دیا آور یاس ئیت پنا گار وان الو واں ناک ساز پوش گیر حور چیں ریز سوز زدہ کشا ہوس کار باز دوز شدہ بان حال جگر مند ئیہ طور ینی اپن اوٹ ئیے گان باز رنگ ئیس انگ کدہ داد نما یدہ وری کار لوش دار پنا تاؤ ساز کدے ابٹ یاب گیر زنی گری وار مندی بازی داں کوش کدہ سرا زدی شکن بند خیز ئیہ گیو انہ ئیں سرا بخت یلا اسی ویں پتی گاہ یلے دان ینے اتے دست راز یوں تری طلب دلی ربا فام یار بخش
4	ئیوں نزاد منڈی یالی ٹائی آیند دستی آگیں خورہ بائی پسندی زدگی نویس گوئی گیری وائی کراں گراں آباد آمےز اتوں نامے ارتے وائز بندی خوری پوشی نامی دانی نشین ئیت سازی یناس کاری تائی آزما وانا خانہ ٹھاک تانی کاری گاری پنات پنیت پرست پذیر پسند یچوں شکنی ستان نامہ ئیاں جاتی یانا گوار دوست خواہ یابی فراز آزاد پناہ گانہ انتا اننا انتی گزار ٹاتی ائزڈ یتیں یتوں زادہ اروں ئگان ئگاں فہمی ہٹوں یاری یاتی ترین شناس آیند زدگی تامہ ورزی پرور آوری فروش بینی مقدس کدوں فشاں خوان گداز ناکی نشیں کردہ جوئی خیزی گزیں خواں داری رساں فکری خوار طراز گسار کوشی سوزی آلود گیوں نفسی دیدہ نگار زدوں چاری سنجی بختی رنگی وایا سویں ٹانہ گردی پراں ازمز کیسز نواز آمیز آموز کناں نگیں بندی گاہی خانے وانے
5	گواری بیانی آرائی کنندہ گزراں دوراں آزادی شکنوں نگاہی یافتہ انگیز ڈبال پتیاں افشاں ستانی اندیش دہندہ نمائی تہانی بیانی فشانی فروشی سرائی رسائی پروری نشینی گرافز اندوز پرداز بر یوں گر یوں سراؤں آلودہ آمیزی آرائی ماراں حیراں نگراں میراں باراں شبانہ ترانہ بشانہ گرانہ خانوں گاروں دہانی دانوں گواری الدین مندان مثانہ افراز افروز پزیری روائی آفریں گاران انائی ہٹیں اعظم اعل ٰی کشائی زمانہ دارنی زدگان نویسی ماراں رووں خواری شعاری معنوی طرازی بازوں آفرین نشینی مندوں خانوں گاہیں گاہوں خوانی گیروں داروں ثانیہ خارجہ شناسی گساری کشیدہ رسیدہ انداز ورسٹی نگاری سرائی گاہیں گزاری شدگان پرستی
6	گار یاں پسندوں گواریت دار یوں خارجیہ دارانہ طفلانہ پروانے ورز یاں وانزرز گردائی برادری دلنشین گرافیز ٹیگیاں لیواؤں خیز یوں گاہئیں نویسوں گوئیوں سکنوں بوسیوں ساز یاں ریز یاں کار یوں رانیوں آفرینی بہاراں عمراں اختراں نگاراں کارانہ بازانہ جو یانہ روزانہ مندانہ مستانہ غرضانہ افسانہ جرمانہ بیگانہ افزائی خدادادد بردار لابیاں ناترسی یفکیشن دستیوں دستیاں بینیاں خواروں پردازی ابٹیں انگیزی اندوزی باز یوں باز یاں کار یاں ناکیوں ناکیاں کثیرال بندیوں بندیاں خیز یوں خیز یاں پذیروں پذیری پارینہ خوانوں دہندوں دار یاں نویساں نگاروں دانیوں دانیاں دیوانہ گوئیوں گوئیاں اندازی اسیویں سازانہ دوگانہ ترکانہ علمانہ الہانہ متحدال
7	پاشیدگی آہنگیوں وندکریم نخواستہ ناگہانی رسانیوں رسانیاں آرایانہ گزارانہ شامیانہ تاز یانہ سوفیانہ مجرمانہ پرور یاں انگریزی نواز یوں نواز یاں بیانیاں فشانیاں اندازوں تراشیاں پرور یوں خوانیوں پسندانہ حاکمانہ مالکانہ پرستانہ شفاخانہ ظر یفانہ گسترانہ غائبانہ شاعرانہ آرایانہ پنجگانہ آگار یوں غاصبانہ خطیبانہ خودداری شعارانہ کشائیوں کشائیاں نویسیوں وندزادہ آرائیوں آرائیاں رسدخانہ
8	برادر یوں مخالفانہ العلمانہ سپرداری

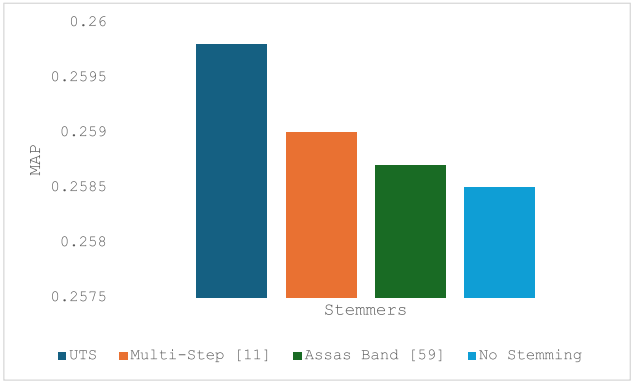

FIGURE 8. Information retrieval results comparison of MAP.

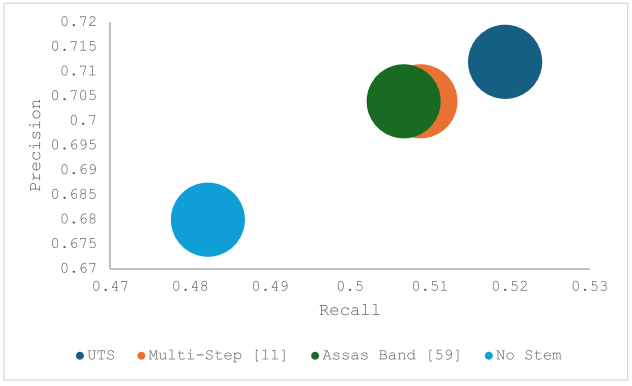

FIGURE 9. Precision recall results of Urdu IR system.

an affix, cause the wrong stemming cases (False Negative). Although we handle such cases by table lookup approach and stem words list, it will depend upon the table lookup and stem words list entries. In the case of text corpus, the accuracy is low as compared to words corpus (see figure 4) due to the improper tokenization and identification of proper nouns, for instance, the Urdu sentence محمدحارث حسین تعلیم یافتہ لڑکاہے [Mohammad Haris Hussain taleem Yafta larka hai /Mohammad Haris

TABLE 18. List of tokenization markers.

Symbol	Unicode	Description	Symbol	Unicode	Description
{	007B	Left curly brackets	A	0041	English capital letter A
}	007D	Right curly brackets	B	0042	English capital letter B
(	0028	Left parenthesis	C	0043	English capital letter C
)	0029	Right parenthesis	D	0044	English capital letter D
[	005B	Left square bracket	E	0045	English capital letter E
]	005D	Right square bracket	F	0046	English capital letter F
>	003D	Greater than sign	G	0047	English capital letter G
<	003C	Less than sign	H	0048	English capital letter H
-	002E	Full stop	I	0049	English capital letter I
_	005F	Low line	J	004A	English capital letter J
=	003D	Equal sign	K	004B	English capital letter K
+	002B	Plus sign	L	004C	English capital letter L
÷	00F7	Division sign	M	004D	English capital letter M
*	002A	Asterisk	N	004E	English capital letter N
!	0021	Exclamation mark	O	004F	English capital letter O
|	007C	Vertical bar	P	0050	English capital letter P
\	005C	Backslash	Q	0051	English capital letter Q
/	002F	Slash (Solidus)	R	0052	English capital letter R
:	003A	colon	S	0053	English capital letter S
؛	06B	Arabic semi-colon	T	0054	English capital letter T
؟	061F	Arabic question mark	U	0055	English capital letter U
~	007E	Tilde	V	0056	English capital letter V
@	0040	At the rate sign	W	0057	English capital letter W
#	0023	Number sign	X	0058	English capital letter X
$	00A4	Currency sign	Y	0059	English capital letter Y
%	0025	Percentage sign	Z	005A	English capital letter Z
&	0026	Ampersand sign	a	0061	English small letter a
^	005E	Circumflex accent	b	0062	English small letter b
-	002D	Hyphen -minus	c	0063	English small letter c
،	060C	Arabic comma sign	d	0064	English small letter d
;	003B	Semicolon	e	0065	English small letter e
“	201C	Left double quotation mark	f	0066	English small letter f
”	201D	Right double quotation mark	g	0067	English small letter g
‘	2018	Left single quotation mark	h	0068	English small letter h
’	2019	Right Left single quotation mark	i	0069	English small letter i
”	0022	Quotation mark	j	006A	English small letter j
!	0021	Exclamation mark	k	006B	English small letter k
-	06D4	Arabic period (Dash)	l	006C	English small letter l
‘	0027	Apostrophe	m	006D	English small letter m
٠	06F0	Arabic digit zero	n	006E	English small letter n
١	06F1	Arabic digit one	o	006F	English small letter o
٢	06F2	Arabic digit two	p	0070	English small letter p
٣	06F3	Arabic digit three	q	0071	English small letter q
٤	06F4	Arabic digit four	r	0072	English small letter r
٥	06F5	Arabic digit five	s	0073	English small letter s
٦	06F6	Arabic digit six	t	0074	English small letter t
٧	06F7	Arabic digit seven	u	0075	English small letter u
٨	06F8	Arabic digit eight	v	0076	English small letter v
٩	06F9	Arabic digit nine	w	0077	English small letter w
0	0030	European digit zero	x	0078	English small letter x
1	0031	European digit one	y	0079	English small letter y
2	0032	European digit two	z	007A	English small letter z
3	0033	European digit three		0x0D	Carriage return
4	0034	European digit four		0x0A	Line feed
5	0035	European digit five		0x0C	Form feed
6	0036	European digit six			
7	0037	European digit seven			
8	0038	European digit eight			
9	0039	European digit nine			

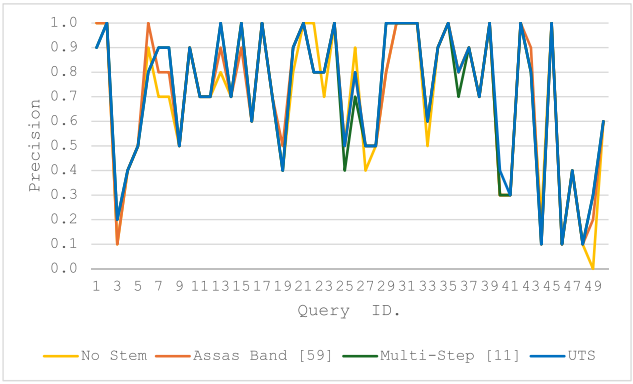

FIGURE 10. Queries wise results of Urdu information retrieval.

Hus- sain is an educated boy] in which compound word تعلیم یافتہ[taleem Yafta/ educated] is not properly extracted due to the proper noun محمدحارث حسین [Mohammad Haris Hussain] and remains un-stemmed or wrong stem is produced. The reason is that in such cases hard space is used as a delimiter, as a result, compound words such as تعلیم یافتہ [taleem Yafta/ educated] become two independent words تعلیم [taleem/education] and suffix یافتہ [yafta]. In this situation, the suffix یافتہ becomes an independent word and is not removed but تعلیم [taleem/education] is stemmed to علم [ilam/knowledge].

Urdu is a scarce resource language that needs different lexical resources for text analysis [71]. Information retrieval performance can improve if the necessary preprocessing resources are built. For example, a stemmer can work well on the word level, but a word segmentation system is still needed for Urdu.

Considering all the results, UTS proves to be the most effective and universal approach for stemming in the morphology-rich language Urdu. UTS excels in IR tasks and inflection removal experiments. MU stemmer [13] consistently delivers good results. However, the Assas-Band stemmer [43] did not perform as efficiently as the other stemmers in our tests.

VIII. CONCLUSION

This paper proposes a novel multi-level inflection and derivation handling stemmer (named UTS) for the Urdu language. According to the best of our knowledge, it is the first stemmer that considers the multi-level inflections in the Urdu language. The evaluation of the UTS shows that considering the multi-level inflections in Urdu stemmers improves the accuracy and performance of the stemming process. As a result, the UTS outperforms the state-of-the-art Urdu stemmers. Given this, the UTS has achieved an accuracy of 94.92% on word corpus and 91.8% on text fragments corpus. We achieved ICF of 55.47%, WSF of 96.62%, CSWF of 95.59%, and AWCF is 54.56. The result of information retrieval highlights that UTS can retrieve relevant information effectively.

Finally, the findings of this research may help to develop NLP tools in the domain of text mining, IR, text summarization, document indexing, spelling checker, parser, thesaurus, and dictionaries. For future works, we plan to investigate more deeply the Urdu word morphology, particularly the word having infixes. The segmentation process can also be enhanced by adding more context-sensitive rules to further improve stemming performance.

APPENDIX A
MOHMIL WORD REDUCTION RULES

See Table 15.

APPENDIX B
URDU PREFIXES

See Table 16.

APPENDIX C
URDU SUFFIXES

See Table 17.

APPENDIX D
TOKENIZATION MARKER

See Table 18.

AUTHOR CONTRIBUTIONS

Conceptualization: Abdul Jabbar, Sajid Iqbal, and Manzoor Ilahi; methodology: Abdul Jabbar; software: Abdul Jabbar; validation: Abdul Jabbar, Sajid Iqbal, Abdullah Abdulrhman Alaulamie, and Manzoor Ilahi; formal analysis: Abdullah Abdulrhman Alaulamie; investigation: Sajid Iqbal and Abdullah Abdulrhman Alaulamie; resources: Sajid Iqbal and Abdullah Abdulrhman Alaulamie; writing—original draft preparation: Abdul Jabbar; writing—review and editing: Abdul Jabbar, Sajid Iqbal, and Manzoor Ilahi; supervision: Sajid Iqbal and Manzoor Ilahi; funding acquisition: Abdullah Abdulrhman Alaulamie. All authors have read and agreed to the published version of the manuscript.

FUNDING

This research received no external funding.

DATA AVAILABILITY STATEMENT

Datasets used in this research are publicly available at the following link:. https://github.com/abduljabbar2017/Urdu-stemmer-dataset

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

[1] A. El Mahdaouy, E. Gaussier, and S. O. El Alaoui, "Should one use term proximity or multi-word terms for Arabic information retrieval?" *Comput. Speech Lang.*, vol. 58, pp. 76–97, Nov. 2019, doi: 10.1016/j.csl.2019.04.002.

[2] C. B. Ali, H. Haddad, and Y. Slimani, "Empirical evaluation of compounds indexing for Turkish texts," *Comput. Speech Lang.*, vol. 56, pp. 95–106, Jul. 2019, doi: 10.1016/j.csl.2019.01.004.

[3] L.-C. Chen, "A novel page clipping search engine based on page discussion topics," *Knowl. Inf. Syst.*, vol. 58, no. 3, pp. 525–550, Mar. 2019, doi: 10.1007/s10115-018-1173-2.
[4] Y. Chen, J. Wang, P. Li, and P. Guo, "Single document keyword extraction via quantifying higher-order structural features of word co-occurrence graph," *Comput. Speech Lang.*, vol. 57, pp. 98–107, Sep. 2019, doi: 10.1016/j.csl.2019.01.007.
[5] S. S. Sahu, D. Dutta, S. Pal, and I. Rasheed, "Effect of stopwords and stemming techniques in Urdu IR," *Social Netw. Comput. Sci.*, vol. 4, no. 5, p. 547, Jul. 2023, doi: 10.1007/s42979-023-01953-4.
[6] Y. A. Alhaj, J. Xiang, D. Zhao, M. A. A. Al-Qaness, M. A. Elaziz, and A. Dahou, "A study of the effects of stemming strategies on Arabic document classification," *IEEE Access*, vol. 7, pp. 32664–32671, 2019, doi: 10.1109/ACCESS.2019.2903331.
[7] A. Razali, S. Mohd, N. Azan, and F. Shahidi, "Stemming text-based web page classification using machine learning algorithms: A comparison," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 1, pp. 570–576, 2020, doi: 10.14569/ijacsa.2020.0110171.
[8] M. Labani, P. Moradi, F. Ahmadizar, and M. Jalili, "A novel multivariate filter method for feature selection in text classification problems," *Eng. Appl. Artif. Intell.*, vol. 70, pp. 25–37, Apr. 2018, doi: 10.1016/j.engappai.2017.12.014.
[9] A. Jabbar, S. Iqbal, M. U. G. Khan, and S. Hussain, "A survey on Urdu and Urdu like language stemmers and stemming techniques," *Artif. Intell. Rev.*, vol. 49, no. 3, pp. 339–373, Mar. 2018, doi: 10.1007/s10462-016-9527-1.
[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 Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 21, no. 2, pp. 1–20, Mar. 2022, doi: 10.1145/3476226.
[11] A. Jabbar, S. Iqbal, A. Akhunzada, and Q. Abbas, "An improved Urdu stemming algorithm for text mining based on multi-step hybrid approach," *J. Experim. Theor. Artif. Intell.*, vol. 30, no. 5, pp. 1–21, May 2018, doi: 10.1080/0952813x.2018.1467495.
[12] M. Ali, S. Khalid, and M. H. Aslam, "Pattern based comprehensive Urdu stemmer and short text classification," *IEEE Access*, vol. 6, pp. 7374–7389, 2018, doi: 10.1109/ACCESS.2017.2787798.
[13] M. S. Husain, F. Ahamad, and S. Khalid, "A language independent approach to develop Urdu stemmer," in *Advances in Intelligent Systems and Computing*. Berlin, Germany: Springer, 2013, pp. 45–53, doi: 10.1007/978-3-642-31600-5_5.
[14] W. Khan, A. Daud, J. A. Nasir, T. Amjad, S. Arafat, N. Aljohani, and F. S. Alotaibi, "Urdu part of speech tagging using conditional random fields," *Lang. Resour. Eval.*, vol. 53, no. 3, pp. 331–362, Sep. 2019.
[15] A. Qureshi, D. Anwar, and M. Awan, "Morphology of the Urdu language," *INTJR*, vol. 1, no. 3, pp. 20–25, Sep. 2012.
[16] M. A. Haq, قواعد اردو. انجمن ترقی اردو, (ہند) نئی دلی, 1996.
[17] A. Deutsch, H. Velan, and T. Michaly, "Decomposition in a non-concatenated morphological structure involves more than just the roots: Evidence from fast priming," *Quart. J. Exp. Psychol.*, vol. 71, no. 1, pp. 85–92, Jan. 2018.
[18] R. L. Schmidt, *Urdu: An Essential Grammar*. Evanston, IL, USA: Routledge, 2005.
[19] D. S. A. Baloch, بنیادی اردو قواعد, First. زبان پاکستان اسلام آباد: مقتدرہ قومی.
[20] S. M. J. Rizvi and M. Hussain, "Analysis, design and implementation of Urdu morphological analyzer," in *Proc. Student Conf. Eng. Sci. Technol.*, Aug. 2005, pp. 1–7, doi: 10.1109/SCONEST.2005.4382901.
[21] A. Parveen. (2015). *Morphological Analysis of Modern Standard Urdu*. [Online]. Available: https://shodhganga.inflibnet.ac.in/handle/10603/52303
[22] R. A. Islam, "The morphology of loanwords in Urdu: The Persian, Arabic and English strands," Ph.D. dissertation, School English Literature, Lang. Linguistics, Newcastle Univ., Tyne, U.K., 2012.
[23] J. B. Lovins, "Development of a stemming algorithm," *Mech. Transl. Comput. Linguist.*, vol. 11, nos. 1–2, pp. 22–31, 1968.
[24] M. F. Porter, "An algorithm for suffix stripping," *Program*, vol. 14, no. 3, pp. 130–137, Mar. 1980, doi: 10.1108/eb046814.
[25] D. R. Chintala and E. M. Reddy, "An approach to enhance the CPI using Porter stemming algorithm," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 3, no. 7, pp. 1148–1156, 2013.
[26] J. Savoy, "A stemming procedure and stopword list for general French corpora," *J. Amer. Soc. Inf. Sci.*, vol. 50, no. 10, pp. 944–952, 1999, doi: 10.1002/(SICI)1097-4571(1999)50:10<944:AID-ASI9>3.0.CO2-Q.
[27] A. G. Jivani and M. Anjali, "A comparative study of stemming algorithms," *Int. J. Comp. Tech. Appl.*, vol. 2, pp. 1930–1938, Oct. 2004.
[28] T. Brychcín and M. Konopík, "HPS: High precision stemmer," *Inf. Process. Manage.*, vol. 51, no. 1, pp. 68–91, Jan. 2015, doi: 10.1016/j.ipm.2014.08.006.
[29] P. Singh and P. K. Bhowmick, "Neural network guided fast and efficient query-based stemming by predicting term co-occurrence statistics," *Social Netw. Comput. Sci.*, vol. 3, no. 3, pp. 1–19, May 2022, doi: 10.1007/s42979-022-01081-5.
[30] M. Bacchin, N. Ferro, and M. Melucci, "A probabilistic model for stemmer generation," *Inf. Process. Manage.*, vol. 41, no. 1, pp. 121–137, Jan. 2005, doi: 10.1016/j.ipm.2004.04.006.
[31] K. Abainia, S. Ouamour, and H. Sayoud, "A novel robust Arabic light stemmer," *J. Exp. Theor. Artif. Intell.*, vol. 29, no. 3, pp. 557–573, May 2017, doi: 10.1080/0952813x.2016.1212100.
[32] H. Alshalabi, S. Tiun, N. Omar, and F. N. Al-Aswadi, "Arabic light-based stemmer using new rules," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 9, pp. 6635–6642, 2022.
[33] J. Atwan, M. Wedyan, and H. Al-Zoubi, "Arabic text light stemmer," *Int. J. Comput. Acad. Res.*, vol. 8, no. 2, pp. 17–23, 2019.
[34] S. Bessou and M. Touahria, "An accuracy-enhanced stemming algorithm for Arabic information retrieval," 2019, *arXiv:1911.08249*.
[35] M. N. Al-Kabi, S. A. Kazakzeh, B. M. A. Ata, S. A. Al-Rababah, and I. M. Alsmadi, "A novel root based Arabic stemmer," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 27, no. 2, pp. 94–103, Apr. 2015, doi: 10.1016/j.jksuci.2014.04.001.
[36] B. Abuata and A. Al-Omari, "A rule-based stemmer for Arabic Gulf dialect," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 27, no. 2, pp. 104–112, Apr. 2015, doi: 10.1016/j.jksuci.2014.04.003.
[37] H. Alshalabi, S. Tiun, N. Omar, E. A. Anaam, and Y. Saif, "BPR algorithm: New broken plural rules for an Arabic stemmer," *Egyptian Informat. J.*, vol. 23, no. 3, pp. 363–371, Sep. 2022.
[38] A. Alnaied, M. Elbendak, and A. Bulbul, "An intelligent use of stemmer and morphology analysis for Arabic information retrieval," *Egyptian Informat. J.*, vol. 21, no. 4, pp. 209–217, Dec. 2020, doi: 10.1016/j.eij.2020.02.004.
[39] O. Aldabbas, G. Kanaan, M. Albdarnah, R. Alshalabi, M. A. Shehab, and N. Mahyoub, "Technique of regular expression for Arabic light stemmer," *Int. J. Adv. Stud. Comput. Sci. Eng.*, vol. 5, no. 11, p. 175, 2016.
[40] M. El-Defrawy, Y. El-Sonbaty, and N. A. Belal, "A rule-based subject-correlated Arabic stemmer," *Arabian J. Sci. Eng.*, vol. 41, no. 8, pp. 2883–2891, Aug. 2016, doi: 10.1007/s13369-016-2029-2.
[41] I. Zeroual, M. Boudchiche, A. Mazroui, and A. Lakhouaja, "Developing and performance evaluation of a new Arabic heavy/light stemmer," in *Proc. 2nd Int. Conf. Big Data, Cloud Appl.* New York, NY, USA: ACM, Mar. 2017, pp. 1–6, doi: 10.1145/3090354.3090371.
[42] A. M. Saeed, T. A. Rashid, A. M. Mustafa, R. A. A.-R. Agha, A. S. Shamsaldin, and N. K. Al-Salihi, "An evaluation of reber stemmer with longest match stemmer technique in Kurdish sorani text classification," *Iran J. Comput. Sci.*, vol. 1, no. 2, pp. 99–107, Jun. 2018, doi: 10.1007/s42044-018-0007-4.
[43] A. M. Mustafa and T. A. Rashid, "Kurdish stemmer pre-processing steps for improving information retrieval," *J. Inf. Sci.*, vol. 44, no. 1, pp. 15–27, Feb. 2018, doi: 10.1177/0165551516683617.
[44] P. Koirala and A. Shakya, "A nepali rule based stemmer and its performance on different NLP applications," 2020, *arXiv:2002.09901*.
[45] A. A. Suryani, D. H. Widyantoro, A. Purwarianti, and Y. Sudaryat, "The rule-based Sundanese stemmer," *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 17, no. 4, pp. 1–28, Dec. 2018.
[46] H. Kaur and P. K. Buttar, "A rule-based stemmer for Punjabi verbs," *Int. J. Adv. Res. Comput. Sci.*, vol. 6, no. 5, pp. 7962–7966, 2019.
[47] H. Kaur and P. K. Buttar, "A rule-based stemmer for Punjabi adjectives," *Int. J. Adv. Res. Comput. Sci.*, vol. 11, no. 6, pp. 15–19, Dec. 2020, doi: 10.26483/ijarcs.v11i6.6665.
[48] M. Shah, H. Shaikh, J. Mahar, and S. Mahar, "Sindhi stemmer for information retrieval system using rule-based stripping approach," *Sindh Univ. Res. J.-SURJ Sci. Ser.*, vol. 48, no. 4, pp. 891–898, 2016.
[49] A. Bimba, N. Idris, N. Khamis, and N. F. M. Noor, "Stemming Hausa text: Using affix-stripping rules and reference look-up," *Lang. Resour. Eval.*, vol. 50, no. 3, pp. 687–703, Sep. 2016, doi: 10.1007/s10579-015-9311-x.

[50] S. Aslamzai and S. Saad, "Pashto language stemming algorithm," *Asia–Pacific J. Inf. Technol. Multimedia*, vol. 4, no. 1, pp. 25–37, Jun. 2015, doi: 10.17576/apjitm-2015-0401-03.
[51] S. P. Meitei, B. S. Purkayastha, and H. M. Devi, "Development of a Manipuri stemmer: A hybrid approach," in *Proc. Int. Symp. Adv. Comput. Commun. (ISACC)*, Sep. 2015, pp. 128–131, doi: 10.1109/ISACC.2015.7377328.
[52] H. Taghi-Zadeh, M. H. Sadreddini, M. H. Diyanati, and A. H. Rasekh, "A new hybrid stemming method for Persian language," *Digit. Scholarship Humanities*, vol. 32, Nov. 2015, Art. no. fqv053, doi: 10.1093/llc/fqv053.
[53] S. A. Khan, W. Anwar, U. Ijaz Bajwa, and X. Wang. (2012). *A Light Weight Stemmer for Urdu Language: A Scarce Resourced Language*. [Online]. Available: http://www.the-comma.com/diacritics.php
[54] R. Kansal, V. Goyal, and G. S. Lehal, "Rule based Urdu stemmer," in *Proc. COLING 2012 Demonstr.*, Dec. 2012, no. December, pp. 267–276.
[55] V. Gupta, N. Joshi, and I. Mathur, "Design & development of rule based inflectional and derivational Urdu stemmer 'Usal,'" in *Proc. Int. Conf. Futuristic Trends Comput. Anal. Knowl. Manage. (ABLAZE)*, Feb. 2015, pp. 7–12, doi: 10.1109/ABLAZE.2015.7154958.
[56] M. Ali, S. Khalid, M. Haneef, W. Iqbal, A. Ali, and G. Naqvi, "A rule based stemming method for multilingual Urdu text," *Int. J. Comput. Appl.*, vol. 134, no. 8, pp. 10–18, Jan. 2016, doi: 10.5120/ijca2016907784.
[57] M. Ali, S. Khalid, and M. Saleemi, "A novel stemming approach for Urdu language," *J. Appl. Env. Biol. Sci.*, vol. 4, no. 7S, pp. 436–443, 2014.
[58] Z. Hussain, S. Iqbal, T. Saba, A. S. Almazyad, and A. Rehman, "Design and development of dictionary-based stemmer for the Urdu language," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 15, pp. 3560–3569, 2017.
[59] Q.-U.-A. Akram, A. Naseer, and S. Hussain, "Assas-band, an affix-exception-list based Urdu stemmer," in *Proc. 7th Workshop Asian Lang. Resour. (ALR7)*, 2009, pp. 40–47, doi: 10.3115/1690299.1690305.
[60] N. Bölücü and B. Can, "Unsupervised joint PoS tagging and stemming for agglutinative languages," *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, vol. 18, no. 3, pp. 1–21, Sep. 2019, doi: 10.1145/3292398.
[61] S. Khan, W. Anwar, U. Bajwa, and X. Wang, "Template based affix stemmer for a morphologically rich language," *Int. Arab J. Inf. Technol.*, vol. 12, no. 2, pp. 146–154, Mar. 2015.
[62] R. Kansal, V. Goyal, and G. S. Lehal. (2012). *Rule Based Urdu Stemmer*. [Online]. Available: https://www.aclweb.org/anthology/C12-3034.pdf
[63] V. Gupta, N. Joshi, and I. Mathur, "Rule based stemmer in Urdu," in *Proc. 4th Int. Conf. Comput. Commun. Technol. (ICCCT)*, Sep. 2013, pp. 129–132, doi: 10.1109/ICCCT.2013.6749615.
[64] S. R. Sirsat, V. Chavan, and H. S. Mahalle, "Strength and accuracy analysis of affix removal stemming algorithms," *Int. J. Comput. Sci. Inf. Technol.*, vol. 4, no. 2, pp. 265–269, 2013. [Online]. Available: https://www.researchgate.net/publication/280933739
[65] W. B. Frakes and C. J. Fox, "Strength and similarity of affix removal stemming algorithms," *ACM SIGIR Forum*, vol. 37, no. 1, pp. 26–30, Apr. 2003, doi: 10.1145/945546.945548.
[66] C. D. Paice, "An evaluation method for stemming algorithms," in *Proc. SIGIR*, U.K. London, Springer, 1994, pp. 42–50, doi: 10.1007/978-1-4471-2099-5_5.
[67] C. I. Muntean, F. M. Nardini, R. Perego, N. Tonellotto, and O. Frieder, "Weighting passages enhances accuracy," *ACM Trans. Inf. Syst.*, vol. 39, no. 2, pp. 1–11, Apr. 2021, doi: 10.1145/3428687.
[68] P. T. Board, اردو قواعدوانشاء. Lahore: Punjab Textbook Board, 2010.
[69] UEP, گرائمر اردو تخلیق. Unique Education Publisher, Urdu bazar Lahore, 2014.
[70] M. Iqbal, K. Amjad, B. Tahir, and M. A. Mehmood, "CURE: Collection for Urdu information retrieval evaluation and ranking," 2020, *arXiv:2011.00565*.
[71] A. Khattak, M. Z. Asghar, A. Saeed, I. A. Hameed, S. A. Hassan, and S. Ahmad, "A survey on sentiment analysis in Urdu: A resource-poor language," *Egyptian Informat. J.*, vol. 22, no. 1, pp. 53–74, Mar. 2021.
[72] S. Hussain, "Finite-state morphological analyzer for Urdu," M.S. thesis, Dept. Comput. Sci., Center Res. Urdu Lang. Process., Nat. Univ. Comput. Emerg. Sci., Pakistan, 2004.

ABDUL JABBAR received the master's degree in computer science from the Department of Computer Science, Institute of Southern Punjab, Multan. He is currently pursuing the Ph.D. degree with the Department of Computer Science, COMSATS University Islamabad, Pakistan. His research interests include natural language processing, computational linguistics, text mining, and artificial intelligence.

SAJID IQBAL received the Ph.D. degree from the Department of Computer Science, University of Engineering and Technology, Lahore, Pakistan. He is currently an Assistant Professor with the Department of Information Systems, College of Computer. He has published more than 30 papers in local and international journals and conferences. His research interests include medical image analysis, natural language processing, and computer vision.

ABDULLAH ABDULRHMAN ALAULAMIE, photograph and biography not available at the time of publication.

MANZOOR ILAHI received the master's degree in computer science from Gomal University, Dera Ismail Khan, in 1998, and the Ph.D. degree in computer science. After completion of the Ph.D. degree in computer science, he rejoined the Department of Computer Science, COMSATS University Islamabad, as an Assistant Professor, in 2009. He is currently a Professor with the Department of Computer Science, COMSATS University Islamabad. In 2005, he was awarded the COMSATS University Islamabad Scholarship for Ph.D. studies at GSCAS, Beijing, China.