# Challenges in Urdu Text Tokenization and Sentence Boundary Disambiguation

**Zobia Rehman, Waqas Anwar, Usama Ijaz Bajwa**
Department of Computer Science
COMSATS Institute of Information Technology, Abbottabad, Pakistan
{zobiarehman,waqas,usama}@ciit.net.pk

## Abstract

Urdu is morphologically rich language with different nature of its characters. Urdu text tokenization and sentence boundary disambiguation is difficult as compared to the language like English. Major hurdle for tokenization is improper use of space between words, where as absence of case discrimination makes the sentence boundary detection a difficult task. In this paper some issues regarding both of these language processing tasks have been identified.

## 1 Introduction

Urdu is morphologically rich language, spoken by more than 150 million people of the world; either as their mother tongue or as their second language. This language is composed of many different languages, e.g. Arabic, Persian, Turkish, Hindi, Sanskrit, and English. Moreover it adopts new words from other languages. It is a bidirectional language and uses Arabic based orthography. Morphology of Urdu language is influenced by all the languages mentioned above (Riaz, 2007) (Waqas et al., 2006).

Text tokenization is the process of identifying word peripheries in written text. It divides the text into its constituent words (Kaplan, 2005) (Manning et al., 1999). It is a preliminary task for all language processing systems, e.g., machine translation, part of speech tagging, information retrieval, information extraction, grammar checker, and spell checker. All these language processing systems need their input text with definite word boundaries.

Sentence boundary disambiguation is the process of identifying sentence terminating punctuations in written text. It divides the text into its component sentences. Sentence boundary has its own importance in above mentioned language processing systems as well as it is equally important for; text summarization, text paragraphing, parsing, and chunking. These systems need their input text properly alienated into sentences. Tokenization and sentence boundary disambiguation are not easy tasks for Urdu language. Urdu is a complex language with respect to its morphology and nature of its characters. In hand written Urdu text there is no convention to use space for the isolation of words from one another. The native speaker of the language decides about the word boundary by just looking at the shape of characters. Tokenization becomes easy, if there is use of space between words but in the computer typed Urdu text the use of space is extremely uneven; as it is used in some specific situations and this conditional use of spaces makes tokenization even more complex (Lehal, 2010). English also has another advantage of case discrimination in characters. This case discrimination is helpful in identifying sentence boundaries. But Urdu also lacks the case discrimination, which is the only hint to know the starting point of a sentence.

## 2 Literature review

### 2.1 Segmentation techniques

Numerous tokenization techniques are used for various languages of the world, e.g., rule based techniques (Kaplan, 2005) , statistical techniques (Lehal, 2010) , fuzzy techniques (Shahabi et al, 2007), lexical techniques (Wu et al., 1994) (Xing et al., 2008) , and feature based techniques (Meknavin, 1997). Significant work has been done for Arabic (Attia, 2007) and Persian language (Shamsford et al., 2009) also. In (Lehal, 2010) Space omission issues of Urdu script have been addressed and resolved using bilingual corpora and statistical word disambiguation techniques.



*Proceedings of the 2nd Workshop on South and Southeast Asian Natural Language Processing (WSSANLP), IJCNLP 2011,* , pages 40–45,
Chiang Mai, Thailand, November 8, 2011.

### 2.2 Techniques for sentence boundary detection

The task of sentence boundary disambiguation is performed for numerous languages. Although few of them are Arabic script languages, written from right to left, but still no significant work has been done for Urdu sentence boundary disambiguation.

Various techniques have been used for different languages, e.g., rule based techniques (Dincer et al., 2004), collocation identification (Kiss et al., 2006), regular expressions (Walker et al., 2001), finite state models (Rezaei, 2001), heuristic rules, artificial neural network models (Palmer et al., 1994) and part of speech tagging (Mikheev, 2000).

## 3 Issues of text tokenization in Urdu

There is no concept of the space in hand written Urdu text. A native speaker of this language can understand and identify where a word ends and from where a new word starts. But a machine can not behave like a native speaker of the language and can not interpret a text without obvious boundaries of words. If there are two words "آبی" (water) and "پرندے" (birds), in hand written text a speaker can distinguish between the two words but if these two words are written in any computer application then they must be separated with space so that machine can understand them as two different words, e.g., "آبی پرندے" (water birds). To avoid space character, a unique Urdu character known as Zero Width Non-Joiner is used. It just separates the two words without any space between them, e.g., "آبی‌پرندے" (water birds). If space or zero width non joiner are not used then it will consider them a single word, e.g., "آبیپرندے" (water birds), which is not understandable even for the native speaker of the language.

There are two types of characters in Urdu; Joiner and non joiner characters. Inter word space is only used when a word ends with a joiner character. If the word ends with a non joiner character then this space is rarely used. So to properly tokenize the Urdu text, it is needed to manipulate space between words.

Tokenization issues can be mainly divided into following two categories;

- Space inclusion issues
- Space exclusion issues

### 3.1 Space inclusion issues

When words are written in a way without space between them, then it is needed to insert space between them, so that machine can understand their boundaries. There are many languages in the world, in which words are written without any space. This issue is not easy to resolve as there are numerous ways to insert space between the words. Moreover every way conveys different context of the text.

In Urdu, space insertion is needed in following two cases:

- When word ends with non joiner character.
- When zero width non joiner (ZWNJ) is used between two words.

#### 3.1.1 Word ending at non joiner

Characters given in following table are known as non joiner or separator characters in Urdu.

| ا د ڈ ذ ر ز ژ ژ و ے |
|---|

Table 1. Non joiner characters in Urdu

These characters have the specialty that they can only acquire final shape and can not adopt initial or medial shapes. Any joiner character can be attached at their start but they can not be attached at the start of the joiner character. When a word ends with such a non joiner then space is not inserted after it, as for a native speaker there will be no ambiguity to distinguish it from other words (Naim, 1999) (Siddiqi, 1971). Consult Table 2. for such examples

| اسدشہرسےباہرجاپہنچا (I) | اسد شہر سے باہر جا پہنچا (II) |
|---|---|
| Asad reached out of the city. | |

Table 2. Words ending at non joiners

In example (I) words are written without inter word space and in (II) words are written with space at the end of each word. It is obvious that all the words end at non joiner that's why in examples, I and II the sentence gives the same meanings. Native speaker can understand that both of the examples have same words but example (I) is considered by machine as a single vague word.

It is a major issue how to tokenize a string if it has more than one possible combination. Native speaker can identify the discrete words in

this case also by looking at surrounding words but for machine it is impossible.

### 3.1.2 Use of ZWNJ between two words

Zero width non joiner is used between two words when it is needed to separate them from each other. But ZWNJ does not help to distinguish between word boundaries. It just helps to separate them visually. For example "**پرانی‌سڑک**" (old track), in it both words are separated by an additional ZWNJ character.

| |
|---|
| پرانیسڑک (old track)<br>(Words without space or ZWNJ) |
| پرانی سڑک (old track)<br>(Words separated by space) |
| **پرانی‌سڑک** (old track)<br>(Words separated by ZWNJ) |

Table 3. ZWNJ between words

Tokenizer is also responsible to remove this ZWNJ and insert space instead of it so words can be literally separated.

### 3.2 Space exclusion issues

Space exclusion is another issue of text tokenization. The space that is used to separate the words, some times occurs between words, collectively giving the single meaning. During tokenization these words need to be assigned single boundary. Therefore the space between such words is needed to be excluded.

In following cases this space should be neglected while assigning boundaries to words:

- Compound words
- Reduplication
- Affixation
- Proper nouns
- English words
- Abbreviations and Acronyms

### 3.2.1 Compound words

In Urdu there are following categories of compound words with respect to their formation (Sproat, 1992) (Schmidt, 1999) (Javed, 1985):

- AB formation
- A-o-B formation
- A-e-B formation

It is needed to treat them as a single word as these different combinations form a single word.

#### 3.2.1.1 AB formation

In AB formation two roots or stems join together to form a semantically single word. When first word in the compound unit, ends with a non joiner then it is rare to have a space between them, e.g., "کھاتاپیتا" (well-off) but if it ends with a joiner then space is inserted after it. During tokenization this space must be neglected and these words should be assigned a single boundary (Sproat, 1992). See Table 2. for such examples

| |
|---|
| محنت مشقت (hard work) |
| روٹی کپڑا(basic needs of life) |
| ماں باپ (parents) |

Table 4. AB formation of compound words

#### 3.2.1.2 A-o-B formation

In A-o-B formation two roots or stems are linked to each other with the help of a linking morpheme 'و' and make a single semantic unit. If the first morpheme ends at a non joiner then there is no need to insert space between it and linking morpheme, e.g., "درودیوار" (boundary). But if the first morpheme ends with joiner then space is used between it and the linking morpheme. So the tokenizer must neglect this space and consider the compound unit as a single token (Sproat, 1992).

Consider the following examples in Table 5. In it space is used before and after the linking morpheme. Without the space these words will not be understandable even for the native speaker but use of the space brings hurdle, if it is needed to assign a single boundary to these words.

| |
|---|
| عزت و حرمت (honor) |
| نظم و ضبط (discipline) |
| امن و امان (law and order) |

Table 5. A-o-B formation of compound words

#### 3.2.1.3 A-e-B formation

In A-e-B formation "e" is the linking morpheme which shows the relation between A and B. morpheme "e" is represented in Urdu by diacritic "ِ". But before tokenization all diacritics are removed and "ِ" is replaced by space (Sproat, 1992). See the examples in Table 6.

| |
|---|
| وزیر اعظم (prime minister) |
| طالب علم (student) |
| حد نظر (scene limit) |

Table 6. A-e-B formation of compound words

Words of this type must be assigned a single word boundary by excluding the inter word space between them.

### 3.2.2 Reduplication

Reduplicated words must also be considered a single semantic unit and if there is a space between them, then it should be excluded in order to assign a single boundary to reduplicated words (Sproat, 1992).

| | | |
|---|---|---|
| دن بدن (day by day) | دھوم دھام (pomp & show) | اٹھ اٹھ (getup) |
| حرف بحرف (character by character) | صبح صبح (early morning) | روٹی ووٹی (bread) |

Table 7. Reduplication of words

In the examples in Table 7, all the reduplicated words are separated by space. Tokenizer is responsible to neglect this space and mark them as a single word.

### 3.2.3 Affixation

Affixes are commonly used in Urdu. Both prefixes and suffixes are used in it. Whenever any affix (prefix or suffix) or stem are individual morphemes and prefix ends with a joiner then space is inserted between the prefix and the stem. Similarly if the stem ends with a joiner then space is inserted between stem and suffix. But they are single semantic units so these must be encapsulated in a single boundary by excluding the space between stem and affix (Sproat, 1992) (Platts, 2002). See the examples of prefixes in Table 8.

| | |
|---|---|
| خوش اخلاق (polite) | خوش نصیب (lucky) |
| بیش قیمت (expensive) | ان تھک (hard work) |

Table 8. Prefixation

See the examples of suffixes given in Table 9.

| | |
|---|---|
| آلہ کار (apparatus) | حیرت انگیز (amazing) |
| سرمایہ کاری (investment) | شادی شدہ (married) |
| غلط فہمی (misunderstanding) | دہشت ناک (fearful) |

Table 9. Suffixation

### 3.2.4 Proper nouns

Most of the time proper names are divided into first name and last name or into first name,second name and last name (Schmidt, 1999). It is often seen that space is used between these parts but this space should be excluded, so that a name with all its parts can become a single token (Sproat, 1992). Proper noun examples are given in Table 10.

| | |
|---|---|
| سعودی عرب (Saudi Arabia) | حسن علی (Hassan Ali) |
| اسلام آباد (Islamabad) | صالح بانو (Sawliha Bano) |
| جنوبی افریقہ (South Africa) | زینب نور (Zainab Noor) |

Table 10. Proper nouns containing more than one constituent

### 3.2.5 English words

Some of the English words are used in Urdu. These words are often composed of more than one morpheme. When first of these morphemes, written in Urdu ends with a joiner character then space is used between them. This space should be neglected by the tokenizer to assign these words a single boundary (Sproat, 1992). Such examples are given in Table 11.

| | |
|---|---|
| ٹیلی کمیونیکیشن (telecommunication) | ٹیسٹ میچ (test match) |
| نیٹ ورک (network) | میڈیکل سنٹر (medical center) |
| فٹ بال (football) | ایش ٹرے (ash tray) |

Table 11. Words of English language commonly used in Urdu

### 3.2.6 Abbreviations and acronyms

English abbreviations are used in Urdu, in the form of pronunciation of English characters, written in Urdu, with space between each character's pronunciations. These abbreviations behave as a single word. If these are followed by any name then along with the name they form a single unit (Sproat, 1992). Abbreviation and acronym examples in Urdu are given in Table 12.

| | |
|---|---|
| پی ایچ ڈی (PhD) | ایم قریشی (M.Qureshi) |
| این ایل پی (NLP) | اے کے شاہ (A.K. Shah) |

Table 12. English abbreviations

## 4 Issues of Urdu sentence boundary disambiguation

According to linguists a sentence is an expression. It is a collection of words that conveys a complete thought and contains a subject and predicate. Subject is usually a single word or several words; noun or pronoun. It tells about what or whom the sentence is concerned. Predicate is a verb; it tells what the subject is doing or being in the sentence. In the simple most Urdu sentence the subject comes first, then predicate and finally the verb; whereas the object and the predicative nouns come in the middle of the sentence (Platts, 2002).

In Urdu language sentence boundary disambiguation, challenges arise due to its certain properties such as: absence of capitalization and the use of punctuation marks in abbreviations and acronyms. In English, characters can be written in upper and lower case and the difference in characters case is helpful in identifying the sentence boundaries. There is a convention in English language that if a period is followed by a word starting with capital letter then it has maximum probability to become a sentence marker. But in Urdu there are no case discriminations to indicate the start of the sentence

Punctuations like '-', '.', '؟' and '!' are used as sentence terminators and these can also be used inside the sentence; e.g., in Urdu text '-' is used to describe range between two values, in dates, part of abbreviation, and also as the line breaker. Examples for such cases are given in Table 13.

| |
|---|
| احمد پانچ – چه سال شہر سے باہر رہا- روزگار کے حصول کے لیے اسے دوردراز کے علاقوں کا سفر کرنا پڑا۔<br>(Ahmad was out of the city for five to six years. For the sake of job he had to travel far and wide.) |
| ۰۸-۱۰-۲۰۰۵ کی صبح پاکستان میں زلزلے کے شدید جھٹکے محسوس ہوئے ۔<br>On 08-10-2005, sever earthquake jolts had been felt in Pakistan. |
| یو۔ ایس۔ اے۔ کی معیشت پچھلےدو سالوں میں بہت ہی بـر طرح متاثر ہوئ ۔<br>The economy of the U.S.A. has been badly affected since previous two years. |

Table 13. Use of (-) at different locations in an Urdu sentence

Full stop or '.' is also used as sentence terminator in Urdu script as well as the decimal symbol as shown in Table 14.

| |
|---|
| ریکٹرسکیل پہ زلزلے کی شدت ۷. ۸ ریکاڈ کی گیی.<br>Intensity of the earthquake was 7.8 on Richter scale. |

Table 14. Use of (.) at different locations

If there is punctuation inside the Urdu text then by just considering the characters of its surrounding words, it can not be decided that either a given punctuation is sentence terminator or not. Consult table 15. for such examples

| |
|---|
| واہ! کیا کمال کی جگہ ھے-<br>Wow! What a wonderful place.) |
| وہ چلایا،" میری مدد کرو-"<br>(He Screamed, "Help me.") |
| کیوں؟ اس نے ایسی کیا غلطی کر دی؟<br>(Why? What did he do wrong?) |

Table 15. Ambiguity in sentence boundary due to punctuations

Obviously in the above cases it is difficult for the machine to isolate the punctuations from sentence termination behavior.

## 5 Conclusion and Future work

In this paper issues are described for Urdu text tokenization and sentence boundary disambiguation. In hand written Urdu text, words are written in continuation without any space between them. But computer text files demand a separator, whenever a word ends with joiner character. Without any separator, word of this sort will join itself to next word resulting into an indefinite word that is not understandable even for the native speaker of the language. Demand of this separator is satisfied by inserting space character or zero width non joiner after the words ending with joiner characters. On the other hand words ending at non joiners are not followed by any space character or zero width non joiner. In short this intricate job is concerned to manipulate spaces between words, so that machine can demarcate their boundaries. Different statistical and rule based techniques have been applied on the different languages of the word, which are even much more complex than Urdu language, to solve their segmentation issues. In future we will target some of these techniques along with hand

crafted dictionaries of Urdu compound words, affixations and some commonly used English words in Urdu script.

Sentence boundary disambiguation has its own challenges for Urdu. This task is easier to some extent in the languages with upper and lower case character discrimination. As in English there is convention that a period followed by a word starting with an upper case letter, has maximum probability to be a boundary marker. But in Urdu, the language without case discrimination, it is difficult to find the punctuations showing the behavior of sentence boundary. In future we are aimed to solve these issues by using part of speech information of each word followed by any putative sentence boundary. This information can be helpful to know that either the current word should be followed by a sentence terminator or not.

## References


Attia, M. A. 2007. *Arabic tokenization system*, Proceedings of the 2007 Workshop on Computational Approaches to Semitic Language, 65 – 72.

Dincer B. and Karaoglan B. 2004. *Sentence Boundary Detection in Turkish*, *Advances in Information Systems*, Springer Berlin, pp. 255-262.

Javed I. 1985. *New Urdu Grammar.* Advance Urdu Buru New Dehali.

Kaplan. 2005. *Method of Tokenizing Text*, *Inquiries into Words, Constraints And Contexts.*

Kiss T. and Strunk J. 2006. *Unsupervised Multilingual Sentence Boundary Detection*, MIT press, Volume, 32, pp. 485-525.

Lehal G. 2010. *A word segmentation system for handling space omission problem in Urdu script,* WSSANLP, pp. 43-50.

Manning C. Schuetze H. 1999. *Foundations of Statistical Natural Language Processing*. MIT Press Massachusetts.

Meknavin, S. 1997.*Feature-based Thai Word Segmentation*, Proceedings of Natural Language Processing Pacific Rim Symposium, pp. 35 – 46.

Mikheev A. 2000. *Tagging Sentence Boundaries*, Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference, vol 4**,** pp. 264-271.

Naim C. 1999. *Introductory Urdu*. South Asian Language & Area Center University of Chicago.

Palmer D. and Hearst M. 1994. *Adaptive Sentence Boundary Disambiguation*, Proceedings of the fourth conference on Applied natural language processing, Stuttgart, Germany, pp. 73-83.

Platts, J. 2002**.** *A Grammar of the Hindustani or Urdu Language*, Sang-e-Meel Publications, Lahore

Rezaei S. 2001. *Tokenizing an Arabic Script Language, Arabic NLP Workshop at ACL/EACL*, Toulouse, France.

Riaz K. 2007. *Challenges in Urdu stemming- a progress report*, BCS IRSG Symposium.

Ruth L. Schmidt. 1999. *Urdu, An Essential Grammar*, London: Routeledge Taylor & Francis Group.

Shahabi, A. S., Kangaveri, M.R. 2007. *Intelligent processing system*, IFIP International Federation of Information Processing, Springer Boston 2007, Vol. 228/2007, pp. 411- 420

Shamsford, M., Kiani,S., Shahidi,Y. 2009. *STeP-1: Standard text preparation for Persian language*, CAASL3 Third Workshop on Computational Approaches to Arabic Script- Languages.

Siddiqi. 1971. جامع القواعر. Markazi Urdu Board

Sproat, R. 1992. *Morphology and Computation*. The MIT Press.

Walker et al. 2001. *Sentence Boundary Detection: A Comparison of Paradigms for Improving MT Quality*, Machine translation in the information age", pp. 369-372.

Wu, D., Fung, P. 1994. *Improving Chinese Tokenization with Linguistic Filters on Statistical Lexical Acquisition*, Proceedings of the fourth conference on Applied natural language processing, pp. 180 – 181

Waqas A., Xuan W., Lu Li, Xiao-long W. 2006. A Survey of Automatic Urdu Language Processing. International Conference on Machine Learning and Cybernetics, pp: 4489-4494

Xing, H. C., Zhang, X., Dalians, H. 2008. *Using parallel corpora and Uplug to create a Chinease-English dictionary*, Thesis from Royal Institute of Technology.