Indian Languages Requirements for String Search/comparison on Web

Prashant Verma[1]([✉]), Vijay Kumar[2], and Bharat Gupta[2]

[1] WSI, MeitY, New Delhi, India
[2] MeitY, New Delhi, India
{vkumar,bharatg}@meity.gov.in

Abstract. The document formats and protocols that based on character data is mainly prepared for the web. These protocols and formats can be access as resources that contain the various text files that cover syntactic content and natural language content in some structural markup language. In order to process these types of data, it requires various string based operations such as searching, indexing, sorting, regular expressions etc. These documents inspect the text variations of different types and preferences of the user for string processing on the web. For this purpose, W3C has developed two documents Character Model: String Matching and searching that act as building blocks related two these problems on the web and defining rules for string manipulation i.e. string matching and searching on the web. These documents also focus on the different types of text variations in which same orthographic text uses different character sequences and encodings. The rules defined in these documents act as a reference for the authors, developers etc. for consistent string manipulation on the web. The paper covers different types of text variations seen in Indian languages by taking Hindi as initial language and it is important that these types of variations should reflect in these documents for proper and consistent Indian languages string manipulations on the web.

Keywords: W3C · CSS · Unicode · Html · Head · Style · Body · Doctype

1 Introduction and Background

Unicode and ISO jointly defined the Universal character set for character Model. A web documents authored in the world writing system, languages, scripts to be exchanged, read and search through the successful character model. W3C Standard Document Character Model for the World Wide Web-String Matching gives specifications authors, content developers and software developers a general reference on string identity matching and searching on the World Wide Web. The goal of this document is to make web to process and transmit the text in a consistent, proper and clear way. The successful character model permits documents of web works on different writing systems, scripts, and languages on different platforms so that seamless information can be exchanged, read, and searched by the consumers on the web around the world [1].

A string-searching document of W3C covers string-searching operations on the Web in order to allow greater interoperability. String searching refers to matching of natural

language through the "find" command in a Web browser [2]. It is possible to generate the same text with different character encodings. The Unicode allows this mechanism for the identical text. Normalization is the mechanism by Unicode that is usually perform while string search and comparisons. It converts the text to use all pre-composed and decomposed characters. The Unicode provides few chapters on the searching of the string. Out of which Unicode Collation Algorithm contains information on the searching [3].

2   Variations in User Inputs

2.1   Different Preferences by the Users

The Unicode Standard gives different alternatives to define text but requires that both text should be treated identical. In order to improve efficiency, it is recommended that an application will normalize text before performing string manipulation operations such as search, comparisons on the web. The different variations can occur while define Unicode text such that same character used different Unicode code points sequences [4]. This will cause unexpected results while searching and matching of string by the users as both string uses different code points. Additionally in Indian languages, the same text represents two orthographic representations with different encoding. The spelling variations lead to introduce the inappropriate searching results. The different users can use different spellings of the same text, as both spellings are in used. Some examples are shown below:

हिंदी/ɦĩdi/ &हिन्दी/ɦĩdi/ ,मंडी/mɜ̃ɖɪ/ &मण्डी /mɜ̃ɖɪ/,चम्पक /tʃɜ̃pək/ &चंपक /tʃɜ̃pək/

These types of spelling variation may occur in other Indian languages also.

2.2   Keyboard Representation

It is requires by the Unicode to store and interchanged the characters in the same logical order or we can say that order that user typed through the keyboards. It is not always true that in the different keyboard layouts, keystrokes and input characters are same and one to one. It is depends on the type of the keyboard layout. Some keyboards can produce numerous characters from a single key press and some keyboards use different keystrokes to produce one abstract character. It is the limitations of Indian languages that too many characters need to be fit in one single keyboard. This leads to input more complex Indian languages input methods and which makeover keystrokes sequence in character sequences [4].

The Unicode Standard needs that characters can be stored and interchanged in logical order, i.e. roughly corresponding to the order in which text is typed in via the keyboard or spoken. The main limitations of Indian languages is that a limited number of keys can fit on a keyboard. Some keyboards will generate multiple characters from a single key press. In Indian languages, too many characters to fit on a keyboard and must rely on more complex input methods, which transform keystroke sequences into character

sequences. It might be occurs that different character sequences of the same text used by different users from the different keyboard and create issues in string identity matching.

3   Use-Cases

3.1   Text Variation in Syntactic Content Under HTML/CSS and Other Applications

The role of syntactic content in a document format and protocol is to represent the text that defines the structure of the document format and protocol. The different values used to define id, class name in markup languages and cascading style sheets are a part of syntactic content. In order to produce output as desired, we should ensure that the selectors and id or class name should be same. The below example represents id used with different character sequences.

In the above example, the id name defines in the HTML and CSS works on the same character sequences [5]. Gaps will be there if id name uses different character sequences. This is particularly occurs and leads an issue if markup language and the CSS are being handled or maintained by different persons.

Below examples shows the different character sequences as per Unicode Code Charts and different choices by the users on writing the characters as both forms can be written [6].

```
<!DOCTYPE html>
<html>
<head>
<style>
#हिंदी-शैली-ज़ांच{
  text-align: center;
  color: red;
}
</style>
</head>
<body>
<p id=" हिन्दी-शैली-ज़ांच">Text in red color</p>
<p>This paragraph is not affected by the style.</p>
</body>
</html>
```

हिंदी /ɦĩdi/ =0939+ 093F+0902+0926+0940

हिन्दी /ɦĩdi/=0939+ 093F+0928+094D+0926+0940

ज़ांच /zatʃ/ =095B+093E+0902+091A

ज़ांच /zatʃ/=091C+093C+093E+0902+091A

There are two types of variations are seen especially in Indian language i.e. spelling variations and different character sequences. The character sequences should be same in order to get the right results. Therefore, it is important that characters – to-characters should match so that proper string manipulation should be made on the web.

3.2 Implementation of Internationalized Domain Name and Email Addresses

For the benefit of large amount of users, it is required to internationalize the domain name and email addresses. In order to make this happen, there is a neccessity to deal with the various issues pertaining to Indian languages such as spelling and text input variations [7, 8]. The user does not have the knowledge of normalized form; user might be use different character sequences for domain name in Indian languages. So, it is required to implement different types of variations while searching and comparison of the strings so that the web document formats and protocols performed the right string-matching operation and user perceive the results.

3.3 Indian Language Search Operations on the Web

User can search natural language content by using find command on the web. Different Users might use different character sequences of the same text by performing find command. There should be some common mapping and implementation in order to satisfy the users need. There user might expect that typing one character will find the equivalent character in the same script such as in Devanagari script ल that represents DEVANAGARI LETTER LA and ळ that represents DEVANAGARI LETTER LLA etc.

The few examples are shown below:

आवाज़ /aʋaz/: 0906 + 0935 + 093E + 095B

आवाज़ /aʋaz/: 0906 + 0935 + 093E + 091C + 093C

फ़ाँसी /fãsi/: 095E + 093E +0901 + 0938 + 0908

फ़ाँसी /fãsi/: 092B + 093C + 093E + 0901 + 0938 + 0908

चम्पक /tʃə̃pək/ : 091A + 092E + 094D + 092A + 0915

चंपक /tʃə̃pək/ : 091A + 0902 + 092A + 0915

Additionally in Indian languages, some of the text represents two orthographic representations with different encodings. The spelling variations lead to introduce the inappropriate searching results. Some examples are shown below:

हिंदी /ɦĩdi/&हिन्दी /ɦĩdi/ ,मंडी /mə̃ɖi/&मण्डी /mə̃ɖi/ ,चम्पक /tʃə̃pək/ &चंपक /tʃə̃pək/, ठंडा /ʈʰə̃ɖa/&ठण्डा /ʈʰə̃ɖa/, अम्वर /ə̃ʋər/&अंवर /ə̃ʋər/

4   Current Gaps and Requirements for Indian Languages

The above-defined W3C draft Standards specify the requirements while implementing string matching of syntactic content and search of natural language content by using matching rules. The following Indian language requirements need to introduce in the standards in order to perform proper string operations on the web:

1. Different kinds of character variations are not currently reflected in the Standards. Therefore there is a need to address these gaps for proper implementation and reference purpose. These variations have been discussed in the above sections.
2. Need to analysis variations with in the script such as equivalent form of character in the same script as discussed in above sub Sect. 3.3 and Singleton mapping such as ॐ[U + 0950 DEVANAGRI OM] & 🕉[U + 1F549 OM SYMBOL].
3. In addition, it is recommended that Indian languages characters need to be post processed through normalization defined by Unicode for comparison and searching on the web. This leads to the removal of various ambiguities occurs in Indian languages [9]. Unicode specifies the different normalized forms. such as NFD, NFC, NFKC etc. and discussed in Unicode technical report on Normalization forms [10]. It is recommended that NFC is best suitable normalize form for string manipulation on the web.

5   Conclusion

The paper discussed about the different variations in Hindi characters. The all-Indian languages requirements and variations for web search and comparison need to be reflected in the standards for reference and correct manipulation on the Web. The best way is to ensure that the characters should always be processed through normalization so that the user gets the consistent results. The different ways of text might cause unexpected results. Therefore, it is important for authors/developers to take care about the different requirements of Indian languages and for the correct implementation of syntactic and natural language content in the web documents. This paper can also be extended by the investigation of the different types of variations in other Indian languages apart from Hindi.

References

1. Character Model for the World Wide Web: String Matching (2021). https://www.w3.org/TR/charmod-norm/
2. W3C String Searching (2020). https://w3c.github.io/string-search/
3. UNICODE COLLATION ALGORITHM. https://www.unicode.org/reports/tr10/
4. Unicode Consortium. https://home.unicode.org/
5. Normalization in HTML and CSS. https://www.w3.org/International/questions/qa-html-css-normalization/
6. Unicode Devanagari Code Chart (2020). https://unicode.org/charts/PDF/U0900.pdf
7. Internationalization of Domain Names in Indian Languages 2015. https://www.researchgate.net/publication/277593254_Internationalization_of_Domain_Names_in_Indian_Languages, https://www.tandfonline.com/doi/abs/10.1080/03772063.2005.11416421
8. https://www.tandfonline.com/doi/abs/10.1080/03772063.2005.11416421
9. W3C Requirements for String Identity Matching and String Indexing. https://www.w3.org/TR/charreq/
10. Unicode Normalization Forms (2020). http://www.unicode.org/reports/tr15/