International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:5, No:3, 2011

A Framework for Urdu Language Translation using LESSA

Imran Sarwar Bajwa

***Abstract*—**Internet is one of the major sources of information for the person belonging to almost all the fields of life. Major language that is used to publish information on internet is language. This thing becomes a problem in a country like Pakistan, where Urdu is the national language. Only 10% of Pakistan mass can understand English. The reason is millions of people are deprived of precious information available on internet. This paper presents a system for translation from English to Urdu. A module LESSA is used that uses a rule based algorithm to read the input text in English language, understand it and translate it into Urdu language. The designed approach was further incorporated to translate the complete website from English language o Urdu language. An option appears in the browser to translate the webpage in a new window. The designed system will help the millions of users of internet to get benefit of the internet and approach the latest information and knowledge posted daily on internet.

***Keywords*—**Natural Language Translation, Text Understanding, Knowledge extraction, Text Processing

I. Introduction

In our daily life, the natural languages i.e. English, French, Dutch, Chinese, Urdu, Hindi are basically communication tools. Some are local languages and some other languages are global languages. Global languages are equally popular in all parts of the world [1]. English is one of the global languages and it is the most widely learnt and used language in the world. English is the media language as popular media i.e. Internet, television, newspapers, etc use English. In the current age, internet is the major source of information grasping and learning. Conventionally, the websites are designed into one language and that is usually English. Modern software and web-designing requirements are to provide multi-lingual support for maximum assistance to the users [2]. Therefore, it has become essential to provide more than one medium to the user like English language as the global medium and some other languages like Chinese, Japanese, French, German, Arabic, etc as the native languages. In current times, everybody needs a quick and reliable service. So it was needed that there should be some sort of intelligent software to save time and budget for generating Urdu text from English text for native language based websites designing.

This research addresses a problem that relates to the area of web designing. English is one of the foremost languages in the world that is used to create websites. Due to the global acceptance of English language [3], the foremost medium for web designing is English language and it is mostly used in the commercial, cultural, educational and official websites. So many people cannot get benefits of this huge information source due to the unfamiliarity towards the English language. The major perspective of this conducted research was to present a solution for translation of global languages text into native language text. The cyber information in native languages as Urdu, Arabic, Chinese, Japanese, French, German, etc can be more valuable and advantageous for the users. Commercial websites can be more effective by providing there websites in multi-lingual representation by grasping more customers from various communities even those ones who don't know the global languages as English. Orthodox methods of designing websites for multi-lingual support can be expensive in both terms of time and cost as to create a website from scratch is really a time consuming and effort requiring task.

In the next section, the basics of LESSA [3] have been briefly described. In the next section, the processing steps of the translation system have been illustrated. In the next section, some experiments have been described and in the last section the analysis of the designed system is presented and furthermore the conclusion of the research has been illustrated.

II. Literature Review

Natural languages have been an area of interest for researchers for last many decades. In the late nineteen sixties and seventies, so many researchers as Noam Chomsky [5] Chow, C., & Liu, C [6] contributed in the area of information retrieval from natural languages. They contributed for analysis and understanding of the natural languages, but still there was lot of effort required for better understanding and analysis. Some authors concentrated in this area in eighties and nineties as, Krovetz, R., & Croft, W. B [7], Salton, G., & McGill, M [8], Maron, M. E. and Kuhns, J. L [9], Losee, R. M [10]. These authors worked for lexical ambiguity and information retrieval [8], probabilistic indexing [9], data bases handling [10] and so many other related areas. We also have presented a rule based system [11] that is able to extract desired

This research work was conducted in the department of Computer Science & IT, The Islamia University of Bahawalpur and supported in part by the Higher Education Commission, Pakistan. This paper is the continuation of the series of publications under the research project of Natural Language Processing based Software Designing by the Information Processing Research Group (IPRG).

I. S. B. Author is Assistant Professor in the Department of Computer Science & IT, The Islamia University of Bahawalpur. He is regular member IACSIT. (phone: +92 (062)9255466; e-mail: imran.sarwar@iub.edu.pk)

information from the natural language text. The system understands context and then extracts respective information. This model is further enhanced in this research to capture the information from NL text that is further used for automatic OO modeling.

III. System Design and Components

The language translation system converts the cyber information from English language to Urdu. The designed system first understands the English language text and then generates the XML representation of the input website that is in English language. XML is variant from other markup languages as it typically emphasis on descriptive rather than procedural markup (Burns, J., & Madey, G., 2001). The designed system uses this ability of XML and represents a text document in XML format with Unicode support. Unicode representation is often necessary to translate the input texts encoded as Unicode into some smaller or less general encoding scheme. For Unicode representation XML uses a general purpose mechanism called *string substitution* (Woo, H., & Robinson, W. 2002). XML inherits this mechanism from the mother language of all markup languages, SGML which by-default support Unicode representation skeleton (Malaisé V., Zweigen., 2006). In simple terms, this substantiated mechanism allows for the indirect representation of various types of uninformed parts of a manuscript as they can be the single characters, character strings or a whole text document. Besides many other implementation of this mechanism one is to ensure consistency of taxonomy.

A. LESSA

LESSA (Language Engineering Systems for Semantic Analysis) is a natural language processing approach (Bajwa, 2007). That is primarily used to understand and analyze the natural language text. LESSA is based on a rule-based algorithm. This rule-based algorithm has a number of rules that are defined according to the English grammar. LESSA bases on a rule based algorithm that is used for understanding and analyzing the natural languages text. LESSA has following major modules:

B. Tokenization

This module reads the natural language text and converts the text into lexical tokens. These lexical tokens are processing elements of LESSA. These lexical tokens are further passed to the next phase for further processing.

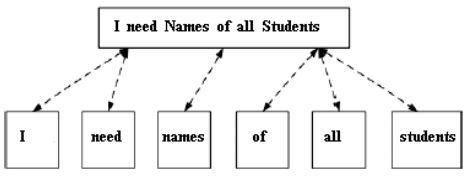

Fig. 1 Lexical analysis of input text string

C. POS Tagging

This module receives the lexical tokens from the previous module and applies a rule-based algorithm on the text. The major function of this module is to identify and extract the various parts of speech in the given text. This phase is also called parts of speech tagging (Bajwa, 2006).

D. Meaning Understanding

This last module understands the semantics of the given test and on the basis of extracted semantics, the object, subject and adverb parts of the sentences are identified. Following is an example to describe the working of LESSA module e.g. "User fills the form."

Typically, LESSA reads this text and performs the lexical analysis to find the parts of speech and then further analyze to find the actor, action and the object in the given course of text. For this example, following is the output.

Lexicons	Phase-I	Phase –II
User	Noun	Actor
fills	Verb	Action
the	Article	-------
form	Noun	Object

Table 01- Lexical analysis of input text string

This is the final output of syntax assessment (Condamines, Anne. R., Josette. 2001) phase and all subject nouns are marked as objects and verbs are marked as actions and all object nouns s are marked as the objects in the scenario. In the above example, there is one actor ‘User’ in the example. Besides the actor, there is also an action ‘fills’ and finally there is an object “form”, on which certain action has been performed .

E. Textual Representation

Input contents provided to the designed system for further processing are in the form of text. Usually, a text is a discriminated into a stream of words. During syntax and semantic analysis the lexical units are divided into many different blocks, of different types or sizes. A simple input in the form of prose text will be divided into its subsequent sections which are further segregated into chapters, paragraphs, and ultimately into sentences. On the other hand, a poetry text might be divided into cantos, stanzas, and lines. These subsections can be divided into super sections as the sequences of prose and verse might be divided into volumes, gatherings, and pages. This multi-facet text requires a hierarchal tree structure representation (Bajwa, 2006)) for better understanding.

F. XML Representation

Text in the conventional format is converted into the simple and consistent mechanism for the markup or identification of textual structure provided by XML (Thomas E., Mark T., Joel W., 2002). According to the definition of textual structure various super and sub category tags can be defined for XML representation. The input text is automatically categorized in

those main and sub tags. Once the normal textual manifested data is converted into the standard XML format, it is more convenient to convert into the desired representation. The super and sub tags for prose scenario can be defined as <chapter>, <paragraph>, <sentence>, <subject>, <object>, <verb>, etc. These are not the final se o tags for such type of information. The abstraction level of these tags can be revised according to the requirements. All these tags also have their corresponding ending tags as shown in the following examples:

```
<chapter>
  <paragraph>
    <sentence>
                Ali is my friend.
    </sentence>
  </paragraph>
</chapter>
```

Example–1: Single XML tags example

In this cataloging, there can be multi <sentence> in a paragraph and multi <paragraph> tags in a <chapter> tag.

```
<chapter>
  <paragraph>
    <sentence>
                Ali is my friend.
    </sentence>
    <sentence>
                He is also my neighbor.
    </sentence>
  </paragraph>
</chapter>
```

Example–2: Multiple nested XML tags example

In the above examples, super to sub categories are represented. In these examples <sentence> tag requires further sub-categorization as it can be divided into parts of a sentence for example subject, helping-verb, verb, object, adverb, prepositions, etc. For example the sentence, 'Mother is cooking meal in kitchen':

```
<sentence>
   <subject>
                Ahmed
    </subject>
    <helping-verb>
                is
    <helping-verb>
    <verb>
                reading
    <verb>
   <preposition>
                a
    <preposition>
   <object>
                book
    <object>
    <preposition>
                in
   <preposition>
   <adverb>
                library.
   <adverb>
</sentence>
```

Example–3: Sub categorization of XML tags

G. Unicode Representation

Unicode is tool which provides the means by which text of all forms and languages can be encoded for various computer applications. The Unicode Standard supports coding of the characters for information interchange (Drouin Patrick., 2004), processing, and display of the miscellaneous languages. In conventional applications, Unicode simply assigns a unique number to each character used by humans in written natural language. Unicode provides 16-bit memory for each character representation. Therefore, the representation of the Asian language script as Urdu is easy to symbolize. XML has built-in ability of supporting Unicode characters hence once the target document is converted into XML format it is quite easy to convert into to desired language.

Each character can be converted into Unicode Transformation Format (UTF) as UTF-8 or UTF-16 or UTF-32 code units. UTF-16 is called also UCS-2 and UTF-32 is also called UCS-4 where UCS stands for Universal Character Set. The designed systems used UTF-16 code. These codes can be represented in both decimal and hexadecimal numbers. For example, the UTF-8 representation of character 'a' is 61 and UTF-16 representation is 0061 (J. M. Zelle and R. J. Mooney, 1993).

There are so many Unicode-based Urdu fonts available as *Umair, Alkatib, Khat-e Naqsh, Aleem, Urdu Web*. As Unicode provides a unique number for every character in Urdu language that is independent of any platform, application or language. Urdu Nastaliq Unicode font uses the Unicode character (K. Christopher, C. Syin, Niu Yun, 2002) coding for displaying various textual scripts. Moreover, Unicode has built in information that the scripts like Arabic and Urdu are written right-to-left. So these language specific characters are automatically placed sequentially, and the Unicode compatible web browsers, like Internet Explorer, Mozilla and Netscape 6 will display them right to left.

IV. System Work Flow

Text input from a web page is read and then its sentence building blocks are separated as shown in the example–3. This phase is performed by syntax analysis which uses a knowledge repository containing a list of prepositions and other related data sets. After Syntax analysis, the sentence level categorization of a sentence (string) is performed. In the last the categorized string of a sentence is represented in to the respective XML tags. These XML tags are globally defined for the unified structure and layout of the resulted XML document (Gómez-Pérez Asunción, F. Mariano, C. Oscar,

2004).

The proposed framework has a profound ability to translate a webpage from English to Urdu language. This framework reads input from a text document which is a webpage in this scenario. To generate XML document from a typical webpage is a multi-step paradigm. The input is read from a webpage and is further processed through various phases. The procedure of reading text from a webpage statement and representation it in XML format is elaborated in the following text. Whole process is segregated into five distinct steps as text input acquisition, text understanding, knowledge extraction, XML document generation and Urdu contents generation.

A. Text input

This module helps to acquire input text from a target webpage. User provides the business scenario in from of paragraphs of the text. These modules read the input text in the form characters and generate the words by concatenating the input characters. This module is the implementation of the lexical phase. Lexicons and tokens are generated in this module.

B. Text Understanding

This phase reads the input from module 1 in from of the lexicons or tokens. These words are categorized into various classes as verbs, helping verbs, nouns, pronouns, adjectives, prepositions, conjunctions, etc.

C. Information Extraction

This phase depicts the various sentence fragments as subject, verb, object, etc, and their respective attributes on the basses of the input provided by the preceding phase. Nouns before helping verb or verb are symbolized as subjects and the nouns after helping-verb or verb are categorized as objects. If the word after a preposition is not noun or pronoun, that is categorized as adverb.

D. XML representation

This phase finally uses the extracted information from the previous modules to actually generate the XML contents. Information of subjects, helping verbs, verbs, objects, adverbs, etc is entrenched into their respective tags. The resulting document contains the Unicode based XML contents of the web document containing simple text.

E. Urdu translation

In the last phase, UTF-16 codes have been used to represent Urdu script characters. All the characters in the XML documents are converted into Urdu characters using Aleem Urdu Unicode font which supports all Urdu script characters from U+0000 to U+007F (0–127).

V. Conclusion

This research paper demonstrates a newly proposed mechanism that any website in English language can be translated into Urdu language by representing the information in XML format in place of conventional web designing techniques. The information representations in XML format can rally enhance the information storage and retrieval efficiency due to the standardized and structured format of the XML documents. The major issue was the dynamic generation of the XML contents from a simple webpage containing text. After analyzing the text available in an input webpage, pertinent XML tags are generated which are ultimately represented in the Unicode manifestation. The designed system has vigorous and profound ability to read and understand the text. The designed system uses a rule based algorithm for this purpose. If the structural layout of the input webpage is very much complex and not properly designed, accuracy may degrade a little bit. An elegant graphical user interface has also been provided to the user for entering the Input scenario in a proper way and generating the desired user interface design.

Current designed system is fairly specific to only one language as Urdu. There can be various improvements in the conducted research. The functionality of the designed system can be enhanced for other languages as Arabic, Chinese, Japanese, etc to provide multi-lingual support in a single website. On the other hand, there is some periphery of improvements in the algorithms for generating the more accurate and precise XML tags.

References

[1] Bajwa, I.S., Choudhary, M.A. (2006) “A Rule Based System for Speech Language Context Understanding” Journal of Donghua University, (English Edition) 23 (6), pp. 39-42.

[2] Gómez-Pérez Asunción, Fernández-López Mariano, Corcho Oscar, (2004) Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web. Springer.

[3] Drouin Patrick. (2004) "Detection of Domain Specific Terminology Using Corpora Comparison." Proceedings of the Fourth International Conference on Language Resources.

[4] Bajwa, I.S., Hyder, I. (2007), "UCD-Generator - A LESSA Application for Use Case Design", Proceedings of IEEE- International Conference on Information and Emerging Technologies- ICIET, pp-182-187

[5] Chomsky, N. (1965) “Aspects of the Theory of Syntax. MIT Press, Cambridge, Mass, 1965.

[6] Chow, C., & Liu, C. (1968) “Approximating discrete probability distributions with dependence trees”. IEEE Transactions on Information Theory, 1968, IT-14(3), 462–467.

[7] Krovetz, R., & Croft, W. B. (1992) “Lexical ambiguity and information retrieval”, ACM Transactions on Information Systems, 10, 1992, pp. 115–141

[8] Salton, G., & McGill, M. (1995) “Introduction to Modern Information Retrieval” McGraw-Hill, New York., 1995

[9] Maron, M. E. & Kuhns, J. L. (1997) “On relevance, probabilistic indexing, and information retrieval” Journal of the ACM, 1997, 7, 216–244.

[10] Losee, R. M. (1988) “Parameter estimation for probabilistic document retrieval models”. Journal of the American Society for Information Science, 39(1), 1988, pp. 8–16.

[11] Imran Sarwar Bajwa, Riaz-Ul-Amin, M. Asif Naeem, Muhammad Nawaz (2006), “Web Information Mining Framework using XML Based Knowledge Representation Engine, Proceedings of 2nd International Conference on Software Engineering, 2006, Lahore, Pakistan

[12] J. M. Zelle and R. J. Mooney, (1993), “Learning semantic grammars with constructive inductive logic programming”, in: Proceedings of the 11th National Conference on Artificial Intelligence (AAAI Press/MIT Press, Washington, D.C.) , pp. 817–822.