Design and implementation of an efficient DeConverter for generating Bangla sentences from UNL expression

Conference Paper · June 2015

DOI: 10.1109/ICIEV.2015.7334006

CITATIONS
0

READS
63

4 authors:

Dr. Aloke Kumar Saha
University of Asia Pacific
27 PUBLICATIONS **54** CITATIONS
SEE PROFILE

M. Firoz Mridha Ph. D.
Bangladesh University of Business and Technology (BUBT)
60 PUBLICATIONS **108** CITATIONS
SEE PROFILE

Molla Rashied Hussein
University of Asia Pacific
14 PUBLICATIONS **21** CITATIONS
SEE PROFILE

J. K. Das
Jahangirnagar University
26 PUBLICATIONS **61** CITATIONS
SEE PROFILE

Design and Implementation of an Efficient DeConverter for generating Bangla Sentences from UNL Expression

Aloke Kumar Saha
Computer Science & Engineering
University of Asia Pacific
Dhaka, Bangladesh
aloke71@yahoo.com

Md. Firoz Mridha
Computer Science & Engineering
University of Asia Pacific
Dhaka, Bangladesh
mdfirozm@yahoo.com

Molla Rashied Hussein
Computer Science & Engineering
University of Asia Pacific
Dhaka, Bangladesh
mrh.cse@uap-bd.edu

Jugal Krishna Das
Computer Science & Engineering
Jahangirnagar University
Savar, Dhaka, Bangladesh
cedas@juniv.edu

***Abstract*—In this paper, the design and implementation of Bangla DeConverter for DeConverting Universal Networking Language (UNL) expressions into the Bangla Language is propounded. The UNL is an Artificial Language, which not only facilitates the translation stratagem between all the Natural Languages across the world, but also proffers the unification of those Natural Languages as well. DeConverter is the core software contrivance in a UNL system. The paper also focuses on the Linguistic Analysis of Bangla Language for the DeConversion process. A set of DeConversion rules have been burgeoned for converting UNL expression to Bangla. Experimental result shows that these rules successfully generate correct Bangla text from UNL expressions. These rules can currently produce basic and simple Bangla sentences; however, it is being aggrandized to superintend advanced and complex sentences.**

Keywords- DeConverter, EnConverter, Machine Translation, UNL.

I. INTRODUCTION

In this era of Information Technology (IT), World Wide Web (WWW) has become the nucleus of essential information. However, a large amount of resources is still beyond the reach of a significant portion of society just because of the man-made Language Barrier. There is a great need to translate digital contents which include but not limited to Websites, Blogs, Online News Portal, E-books, E-Journals and E-mails into the Native Language for overwhelming that Language Barrier. In this multilingual milieu, Machine Translation (MT) is considered as an important tool to unshackle the cordoned mankind. UNL based MT (developed with an interlingua-based approach) is also an effort in this approach. UNL program was primarily launched back in 1996 in the Institute of Advanced Studies (IAS) of United Nations University (UNU), Tokyo, Japan [1], and it is currently supported by the Universal Networking Digital Language (UNDL) Foundation, an autonomous organization founded as an extension of that UNL program afterwards in 2001 with a Head-Quarter (HQ) situated at Geneva, Switzerland [2]. The approach in UNL pertains to the development of the EnConverter and the DeConverter for a Natural Language. EnConverter is used to convert a given sentence into a Natural Language to an equivalent UNL expression, and DeConverter is used to do the vice versa, i.e. to convert a given UNL expression to an equivalent Natural Language sentence. A UNL system has the potential to knock down Language Barriers across the world with the development of optimal $2n$ components, whereas the traditional approaches require shoddy $n(n-1)$ components, where n is the number of Languages.

In this paper, design and development of a Bangla DeConverter has been accorded by accentuating the DeConversion rules and Semantic ambiguity of the Bangla DeConverter. Syntactic alignment is the process of defining arrangements of words in target output. This phase plays a vital role in the accuracy of the generation process.

II. UNL SYSTEM AND ITS STRUCTURE

The UNL system consists of two core tools, namely EnConverter and DeConverter, which are used for the particular Natural Language Processing (NLP), a major branch of Artificial Intelligence (AI). The process of converting a source Language, i.e. Natural Language expression into the desired UNL expression is referred to as EnConversion, and the process of

converting a UNL expression into a target or destination Language, i.e. the desired Native Language expression is referred to as DeConversion. The EnConverter and DeConverter for a Language form a Language Server that may reside inside the Internet. Both the EnConverter and the DeConverter perform their functions on the basis of a set of Grammar rules and a Word Dictionary of Native Language.

UNL representation consists of UNL relations, UNL Attributes (UAs) and Universal Words (UWs). UWs are represented by their English equivalents. These words are listed in the Universal Word Lexicon of UNL knowledge base [6]. Relations are the building blocks of UNL sentences. The relations between the words are drawn from a set of predefined relations [3, 4, 7, 8, 9, 10]. The attribute labels are attached with UWs to provide additional information like Tense, Numbers etc. For example, "করিম কলা খায়" in English "Karim eats banana" can be represented into UNL expression as:

{unl}
agt(eat(icl>consume>do,agt>living_thing,obj>concrete_thing,ins>thing).@entry.@present,karim(icl>name>abstract_thing,com>male,nam<person))
obj(eat(icl>consume>do,agt>living_thing,obj>concrete_thing,ins>thing).@entry.@present,banana(icl>herb>thing))
{/unl}

Here, it should be noted that *agt* is the UNL relation which indicates "a thing which initiates an action"; *obj* is another UNL relation which indicates "a thing in focus which is directly affected by an event"; @entry and @present are UNL attributes which indicate the main Verb and Tense information; and @sg is UNL attribute which indicates the Number information.

III. How DeConverter Works

The DeConverter is a Language Independent Generator (LIG), which provides a Framework for Syntactic and Morphological generation of Native Language. It can convert UNL Expressions into Natural Languages using corresponding Word Dictionaries and sets of DeConversion Rules for converting to the desired Native i.e. Target Languages. A Word Dictionary contains the Information of Words, which correspond to UWs that are included in the UNL Expressions input and the Grammatical Attributes (GAs), which describe the behaviors of the Words. DeConversion Rules describe how to construct a Sentence using the Information from the UNL Expressions input and define in a Word Dictionary. The DeConverter converts UNL Expressions into Sentences of a Target Language following the descriptions of the Generation Rules. The UNL Ontology is also helpful when no corresponding Word for a particular UW exist in that Language. In this case, the DeConverter consults to the UNL Ontology and tries to find a more general UW, of which a corresponding Word exists in its word dictionary and consequently use the word of the upper UW to generate the Target Sentence. The DeConverter works in the following way. First, it transforms the input of a UNL expression, a set of binary relations, into a Directed Graph (DG) structure with Hyper-Nodes called Node-Net. The Root Node of a Node-Net is called Entry Node and represents the Head (e.g. the main Verb) of a Sentence. DeConversion of a UNL Expression is carried out by applying Generation Rules to the Nodes of Node-Net. It starts from the Entry Node, to find an appropriate Word for each Node and generate a Word sequence (a list of words in grammatical order) of the Target Language. In this process, the Syntactic structure is determined by applying Syntactic Rules, and Morphemes are similarly generated by applying Morphological Rules. The DeConversion process ends when all words for all Nodes are found and a Word sequence of the Target Sentence is completed. Fig. 1 shows the structure of the DeConveter. "G" indicates Generation Windows, and "C" indicates Condition Windows of the DeConverter. The DeConverter operates on the Node-List through Generation Windows. Condition windows are used to check conditions when applying a Rule. In the initial stage, on the contrary to the EnConveter [5], the Entry Node of a UNL Expression exists in the Node-List. At the end of DeConversion, the Node-List is the List of all Morphemes, with each as a Node, that are converted from the Node-Net and constitute the Target Sentence [6].

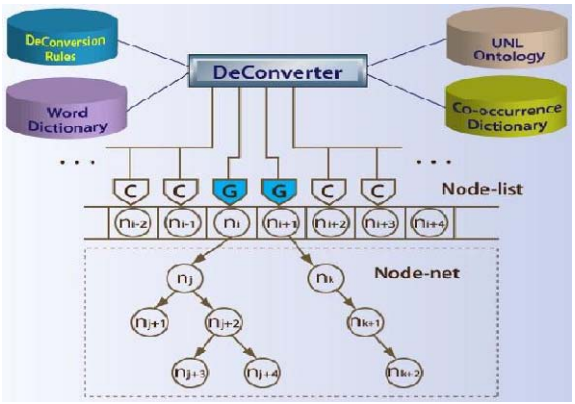

Figure 1. DeConverter structure

IV. DESIGN AND IMPLEMENTATION OF BANGLA DECONVERTER

DeConverter makes use of Language-Independent (LI) and Language-Dependent (LD) components during the generation process [11]. The first stage of DeConverter is the UNL parser, which parses an UNL expression input and builds a Node-Net from that UNL expression input. During the Lexeme selection stage, Bangla Root Words and their Dictionary Attributes are selected for the given UWs in that UNL expression input from the Bangla-UW Dictionary. After that, Nodes are ready for generation of Morphology according to the Target Language in the Morphology phase. In this stage, the Root Words may be changed; i.e., something can be added or removed to obtain the complete sense of Words. The system makes use of Morphological Rules for this purpose. In the Function Word insertion phase, Function Words or Case Markers, such as টি, টা, খানা, খানি, র, এর, য়, তে, থেকে, চেয়ে etc. are inserted to the Morphed Words. These Function Words are inserted in a generated Sentence, based on the Rule-based design in this situation [12]. Finally, the Syntactic Linearization phase is used to define Word order in the generated Sentence, so that the output matches to a Natural Language Sentence [13]. Working of Bangla DeConverter is illustrated with an example Sentence given below:

Bangla Sentence: ছেলেরা মাঠে ফুটবল খেলে। Transliterated Sentence: Chelara mathe football khele.
Equivalent English Sentence: Boys play football in the field.

The UNL expression for example Sentence is given below:

{unl}
agt(play(icl>compete>do,agt>thing).@entry.@present,boy(icl>child>person.@pl)
man(play(icl>compete>do,agt>thing).@entry.@present,football(icl>field_game>thing))
plc(football(icl>field_game>thing),field(icl>tract>thing).@def)
{/unl}

To convert UNL expression to the Bangla Natural Language Sentence, Bangla DeConverter is used. The UNL expression acts as input for the Bangla DeConverter [14]. The UNL parser checks the input UNL expression for errors and generates the Node-Net. The Lexeme selection phase populates the Node-List with the equivalent Bangla Words for the UWs given in the UNL expression input. The populated Node-List is given below:

Node1: Bangla word: খেলে;
UW:play(icl>compete>do,agt>thing).@entry.@present
Node2: Bangla word: ছেলেরা;
UW: boy(icl>child>person.@pl)
Node3: Bangla word: ফুটবল;
UW: football(icl>field_game>thing
Node4: Bangla word: মাঠে;
UW: field(icl>tract>thing).@def)

In the Morphology phase, Morphological Rules are applied to modify Bangla Words stored in the Nodes according to UNL Attributes given in the UNL expression input and Dictionary Attributes retrieved from the Bangla-UW Dictionary [14]. The Nodes are processed by the Morphological Rules.

It is evident that, in the Morphology phase, খেল 'play' is changed to খেলা 'played' and ছেলে 'boy' is changed to ছেলেরা by Morphological Rules. The Function Word insertion phase inserts Function Words in the Morphed Lexicon [15]. Nodes processed by the Function Word insertion phase are given below:

In this phase, Case Markers রা and এ 'in' are added to Node2 and Node4, respectively, according to the Function Word Rule-based insertion. In the Syntactic Linearization phase, one traverses the Nodes in a specific sequence based on the Syntactic Rule-based Linearization for Bangla Language [16].

The sequence for processing of Nodes and the Bangla Sentence generated by this sequence is given below:

Node2 Node4 Node3 Node1
ছেলেরা মাঠে ফুটবল খেলে।

It is evident from the generated Bangla Sentence that the system is able to convert an UNL expression input into a Bangla Natural Language Sentence successfully. The descriptions of different phases of the Bangla DeConverter are given in the following segment.

A. Morphology generation

The System makes use of Generation Rules during this process. These Generation Rules are designed on the basis of Bangla Morphological analysis. There are three Categories of Morphology that have been identified for the purpose of converting a UNL expression to equivalent Bangla Language Sentences. They are: i) Attribute Label Resolution Morphology, ii) Relation Label Resolution Morphology, and iii) Noun, Adjective, Pronoun, and Verb Morphology.
Among these three, major two Morphologies are discussed in details as follow:

i) Attribute Label Resolution Morphology deals with generation of Bangla Words on the basis of

UNL attributes attached to a Node and its Grammatical Attributes retrieved from Lexicon. The Root Words retrieved from Bangla-UW dictionary are modified in this phase, depending on their Gender, Number, Person, Tense, Aspect, Modality, and Vowel ending Information.

ii) Relation Label Resolution Morphology manages the Prepositions in English or Postpositions in Bangla, because Prepositions in English are similar to Postpositions in Bangla. These link Noun, Pronoun, and Phrases to other parts of the sentence. Some insertion of Function Words in generated output depends upon UNL Relation and Conditions imposed on Parent and Child nodes' Attributes in a Relation.

A Rule Base has been prepared for this purpose. For each of 46 UNL Relations, different Function Words are used depending upon the grammatical details of a Target Language [17]. This Rule Base consists of nine Columns. The Attributes whose absence needs to be asserted on the Child Node for firing of the rule are stored. If there is more than one Attribute that needs to be asserted on a given Node for firing of a rule, then they are stored in the Rule Base with the separation of '#' sign. Here, Attributes represent UNL Attributes (obtained from a given UNL expression) or Lexical Attributes (obtained from the Bangla-UW Dictionary) of a Node.

The Rule Base for Function Word insertion is illustrated with an example Rule given below:

agt:null:null:null::@present#V:VINT#@progress#খেল:N#3rd:1st#2nd

Where 'agt' is a UNL relation under consideration, and firing of the given rule will result into insertion of Function Word ে following the Child Node in the generated output, because the Function Word appears in the Fifth Column and the Second, Third, and Fourth Columns contain 'null' in the Rule. The Sixth Column contains '@present#V', which means that the Rule will be fired if the parent of 'agt' relation contains '@present' as its UNL Attribute in the given UNL expression input and has a 'V' as its Lexical Attribute in Bangla-UW Dictionary. The Seventh Column contains 'VINT#@ progress#খেল' which refers to the Attributes whose absence needs to be asserted on the Parent Node for firing of the Rule. It means that the Parent Node should not contain 'VINT' (Intransitive Verb), 'খেল' ('play' Verb) Attributes in the Lexicon or the '@progress' Attribute in the Parent Node of UNL expression. The Eighth Column of the Rule contains 'N#3rd' which refers to the Attribute whose presence needs to be asserted on the Child Node for firing of the Rule; i.e., the Child should have an 'N' (Noun) and '3rd' (Third Person) attribute in the Bangla-UW Dictionary. The Ninth Column contains '1st# 2nd' which refers to the Attribute whose absence needs to be asserted on the Child Node for firing of the Rule. It means that the Child Node should not refer to the First Person or the Second Person in the Sentence [18]. Thus, if the relation 'agt' has a Parent Node with an '@present' and 'V' Attribute, without 'VINT', 'খেল', '@progress', or '@custom' Attribute, or has a Child Node with an 'N' and '3rd' Attribute and without a '1st' or '2nd' Attribute, then Function Word ে will be inserted following the Child Node in generated output [19].

For example, in UNL relation 'agt(play(agt> human, obj>game). @present.@entry, boy (icl> maleperson))' of UNL expression, the Parent Node of Relation 'agt' is 'play(agt>human, obj>game)' having 'V' and '@past' Attribute and without the 'VINT' Attribute in the Lexicon. The Child Node of 'agt' relation is 'boy(icl>male child)' that has 'N' and '3rd' Attribute and does not have '1st' and '2nd' Attributes in the Lexicon. As a result, the firing of Rule will occur and thus the generation of Function Word ে followed by Child Node 'boy(icl> male child)' will be in the generated output as below:

V. Experimental Result and Tessting System

The System has been tested on several UNL Expressions. It has been observed that the System successfully deals with the resolution of UNL Relations and generates Attributes for those Sentences. The System has been tested with the help of UNL Expressions available in the Russian UNL Language Server. The given English sentences were manually translated at Russian Language Server into equivalent UNL Expressions and then those equivalent UNL Expressions were placed into the proposed UNL-Bangla DeConverter mechanism.

Comparative analysis is presented in Table 1 for 5 (Five) Sentences. Accuracy will arise with more tested Sentences and appending Rules. The GUI of Bangla DeConverter is classified into the following three Windows:

(1) Bangla Testing Server
(2) DeConversion
(3) Intermediate output

And Figure 2 shows the Bangla DeConverter Input and Figure 3.shows the Bangla DeConverter Output which is generated by the proposed Bangla DeConverter

Figure 2. Bangla DeConverter Input

Figure 3. Bangla DeConverter Output

Table I. Bangla Sentences generated by the DeConverter with their corresponding UNL expressions Input

Sl.	UNL Expressions generated by the Russian UNL language server	Relations Resolved	Equivalent English Sentence	Bangla Sentences generated by DeConverter
1.	{unl} agt(read(icl>see>do,agt>person,obj>information).@entry.@present.@progress,kerim(icl>name>abstract_thing,com>male,nam<person)) {/unl}	agt	*Karim is reading.*	"করিম পড়িতেছে" *Karim Poritechhe.*
2.	{unl} agt(eat(icl>consume>do,agt>living_thing,obj>concrete_thing,ins>thing).@entry.@present,i(icl>person)) obj(eat(icl>consume>do,agt>living_thing,obj>concrete_thing,ins>thing).@entry.@present,rice(icl>grain>thing)) {/unl}	agt ,obj	*I eat rice.*	"আমি ভাত খাই" *Aami vat khai.*
3.	{unl} agt(write(icl>do,agt>person,obj>concrete_thing,ins>functional_thing).@entry.@past,he(icl>person)) obj(write(icl>do,agt>person,obj>concrete_thing,ins>functional_thing).@entry.@past,note(icl>personal_letter>thing).@indef) ins(write(icl>do,agt>person,obj>concrete_thing,ins>functional_thing).@entry.@past,pen(icl>writing_implement>thing).@indef) {/unl}	agt, ins, obj	*He wrote a Note with a pen.*	"সে কলম দিয়ে একটি নোট লিখেছিল" *Se kolom die ekti note likhechhilo.*
4.	{unl} obj(fly(icl>move>occur,equ>wing,com>air,plt>thing,plf>thing,obj>concrete_thing,plc>thing,ins>thing).@entry.@present,bird(icl>vertebrate>thing).@def) plf(fly(icl>move>occur,equ>wing,com>air,plt>thing,plf>thing,obj>concrete_thing,plc>thing,ins>thing).@entry.@present,nest(icl>retreat>thing).@def) {/unl}	agt, frm, obj	*The bird flies from the nest.*	"পাখিটি বাসা থেকে উড়ে যায়" *Pakhiti basha theke ure jae.*
5.	{unl} aoj(live(icl>be,com>style,aoj>person,man>uw).@entry.@present,we(icl>group).@pl) plc(live(icl>be,com>style,aoj>person,man>uw).@entry.@present,dhaka) {/unl}	aoj, plc	*We live in Dhaka.*	"আমরা ঢাকায় থাকি" *Amra dhakae thaki.*

VI. Conclusion and Future Work

In this paper, a Rule-Based Bangla DeConverter have been proffered. These Rules can currently convert simple UNL Expressions to Bangla Sentences. It is being aggrandized to superintend advanced and complex Sentences. The proposed System has been tested for more than 2000 UNL Expressions. This System achieved accuracy of as good as 89%, which can be marked outstanding in this Field of Study. Moreover, a Web interface has been designed for online DeConversion of the UNL expression to the corresponding Bangla Sentence. It empowers the Bangla Readers to read the sentences in their Local Language, even though those sentences were written initially in a different Language, by having converted through their equivalent UNL expressions presented on the Web. This System will also provide an opportunity for the Researchers to work on MT to explore and expand the UNL beyond its limit to construct the Interlingua Utopia, where Language will no longer be a obstacle for Mankind. Knowledge should not be contained in a jar, rather be let diffuse in an open atmosphere.

References

[1] UNL System. Website Link: *http://www.unl.ru/system.html*. Date of last retrieval: March 14, 2015.

[2] Universal Networking Language Portal. Website link: *http://www.undl.org*. Date of last retrieval: March 14, 2015.

[3] H. Uchida, M Zhu. T.G. Della Senta. *Universal Networking Language*, 2005/6-UNDL Foundation, International Environment House.

[4] H. Uchida, M. Zhu. *The Universal Networking Language (UNL) Specification Version 3.0 Edition 3* , Technical Report, UNU, Tokyo:, 2005/6-UNDL Foundation, International Environment House, 2004

[5] *EnConverter Specification*, Version 3.3, UNL Center/UNDL Foundation, Tokyo 150-8304, Japan 2002

[6] *DeConverter Specification*, Version 2.7, UNL Center, UNDL Foundation, Tokyo 150-8304, Japan 2002

[7] D.M. Shahidullah. *Bangla Baykaron*, Dhaka: Ahmed Mahmudul Haque of Mowla Brothers prokashani, 2003

[8] D. C. Shuniti Kumar. *Bhasha-Prakash Bangala Vyakaran*, Calcutta : Rupa and Company Prokashoni, July 1999, pp.170-175

[9] Humayun Azad. *Bakkotottyo - Second edition*, Dhaka: Bangla Academy Publishers, 1994

[10] D. S. Rameswar. *Shadharan Vasha Biggan and Bangla Vasha*, Pustok Biponi Prokashoni, November 1996, pp. 358-377.

[11] M.N.Y. Ali, J.K. Das, S. M. Abdullah Al Mamun, M. E. H. Choudhury, "*Specific Features of a Converter of Web Documents from Bengali to Universal Networking Language*", International Conference on Computer and Communication Engineering 2008 (ICCCE'08), Kuala Lumpur, Malaysia, pp. 726-731.

[12] M.N.Y. Ali, J.K. Das, S.M. Abdullah Al Mamun, A. M. Nurannabi. "*Morphological Analysis of Bangla words for Universal Networking Language*", International Conference on Digital Information Management (ICDIM), 2008, London, England, pp. 532-537

[13] M.N.Y.Ali, A. M. Nurannabi, G. F. Ahmed, J.K. Das. "*Conversion of Bangla Sentence for Universal Networking Language*", International Conference on Computer and Information Technology (ICCIT), Dhaka, 2010 pp. 108-113

[14] M. Z. H. Sarker, M.N.Y. Ali, J.K. Das, "*Dictionary Entries for Bangla Consonant Ended Roots in Universal Networking Language*", International Journal of Computational Linguistics (IJCL), Volume (3) : Issue (1) :2012, pp. 79-87.

[15] M. Z. H. Sarker, M.N.Y. Ali, J.K. Das, "*Outlining Bangla Word Dictionary for Universal Networking Language*", IOSR Journal of Computer Engineering (IOSRJCE), ISSN: 2278-0661, Sep-Oct. 2012.

[16] M. Z. H. Sarker, M.N.Y. Ali, J.K. Das, "*Development of Dictionary Entries for the Bangla Vowel Ended Roots for Universal Networking Language*", International Journal of Computer Applications (IJCA), *52(19)*, August 2012. Published by Foundation of Computer Science, New York, USA, pp 38-45.

[17] Aloke Kumar Saha Muhammad F. Mridha and Jugal Krishna Das, "*Analysis of Bangla Root Word for Universal Networking Language (UNL)*", International Journal of Computer Applications (IJCA) (0975 – 8887), Volume 89 – No.17, March 2014.

[18] Muhammad F. Mridha, Aloke Kumar Saha and Jugal Krishna Das, "*New Approach of Solving Semantic Ambiguity Problem of Bangla Root Words Using Universal Networking Language (UNL)*", 3rd International Conference on Informatics, Electronics & Vision(ICIEV), 23-24 May, 2014.

[19] Muhammad F. Mridha, Aloke Kumar Saha, Mahadi hasan and Jugal Krishna Das, "*Solving Semantic Problem of Phrases in NLP Using Universal Networking Language (UNL)*" International Journal of Advanced Computer Science and Applications (IJACSA), Special Issue on Natural Language Processing(NLP) 2014.