Universal Networking Language Based Analysis and Generation for Bengali Case Structure Constructs

Kuntal Dey\(^1\) and Pushpak Bhattacharyya\(^2\)

\(^1\) Veritas Software, Pune, India. u2ckuntal@yahoo.com
\(^2\) Computer Science and Engineering Department Indian Institute of Technology, Bombay, India. pb@cs.ee.iitb.ac.in

Abstract. Case structure analysis forms the foundation for any natural language processing task. In this paper we present the computational analysis of the complex case structure of Bengali—a member of the Indo Aryan family of languages—with a view toward interlingua based MT. Bengali is ranked 4\(^{th}\) in the list of languages ordered according to the size of the population that speaks the language. Extremely interesting language phenomena involving morphology, case structure, word order and word senses makes the processing of Bengali a worthwhile and challenging proposition. A recently proposed scheme called the Universal Networking Language has been used as the interlingua. The approach is adaptable to other members of the vast Indo Aryan language family. The parallel development of both the analyzer and the generator system leads to an insightful intra-system verification process in place. Our approach is rule based and makes use of authoritative treatises on Bengali grammar.

1 Introduction

Bengali is spoken by about 189 million people and is ranked 4\(^{th}\) in the world in terms of the number of people speaking the language (ref: http://www.harpercollege.edu/~mhealy/g101ece/intro/clt/cltl/cutop100.html). Like most languages in the Indo Aryan family, descended from Sanskrit, Bengali has the SOV structure with some typical characteristics. A motivating factor for creating a system for processing Bengali is the possibility of laying the framework for processing many other Indian languages too.

Work on Indian language processing abounds. Project Anubad [1] for machine translation from English to Bengali in the newspaper domain uses the direct translation approach. Angalabharati [2] system for English Hindi machine translation is based on pattern directed rules for English, which generates a pseudo-target-language applicable to a group of Indian Languages. In MATRA [3], a web based MT system for English to Hindi in the newspaper domain, the input text is transformed into case-frame like structures and the the target
language is generated by parameterized templates. The MANTRA MT system for official documents uses Tree Adjoining Grammar (TAG) to achieve English Hindi MT (ref: http://www.cdcindia.com/html/about/success/mantra.asp). Project Anusarika [4] is a language access system rather than an MT system and addresses multiple Indian languages. Interlingua based MT for English, Hindi and Marathi [5][6], that uses the UNL, transforms the source text into the UNL representation and generates target text from this intermediate representation. References to most of these works can also be found at http://www.tdlr.mit.gov.in/mat/ach-mat.htm. Other famous MT systems are Pixod [7], Atlas [8], Kant [9], Arct [10], Getv [11], SysTran [12] etc.

The Universal Networking Language (UNL) (http://www.unl.iias.umass.edu) has been defined as a digital meta-language for describing, summarizing, refining, storing and disseminating information in a machine independent and human language neutral form. The information in a document is represented sentence by sentence. Each sentence is converted into a directed hyper graph having concepts as nodes and relations as arcs. Knowledge within a document is expressed in three dimensions:

1. Word Knowledge is expressed by Universal Words (UWs) which are language independent. These UWs are tagged using restrictions describing the sense of the word in the current context. For example, drink(icl > liquor) denotes the noun sense of drink restricting the sense to a type of liquor. Here, icl stands for inclusion and forms an is-a relationship like in semantic nets [13].

2. Conceptual Knowledge is captured by relating UWs through a set of UNL relations [14]. For example,

   Humans affect the environment

   is described in the UNL as

   \[
   \text{agt}(\text{affect}(icl>do).\text{present}.\text{entry}, \text{human}(icl>animal).\text{pl})
   \]

   \[
   \text{obj}(\text{affect}(icl>do).\text{present}.\text{entry}, \text{environment}(icl>abstract\ thing).\text{pl})
   \]

   \[
   \text{agt} \text{ means the agent and obj the object. affect(icl > do), human(icl > animal) and environment(icl > abstract\ thing) are the UWs denoting concepts.}
   \]

3. Speaker's view, aspect, time of event, etc. are captured by UNL attributes. For instance, in the above example, the attribute \text{present} denotes the main predicate of the sentence, \text{present} the present tense and \text{pl} the plural number.

The above discussion can be summarized using the example below

John, who is the chairman of the company, has arranged a meeting at his residence

The UNL for the sentence is

\[
\text{mod(\text{chairman}(icl>post).\text{present}.\text{def}, \text{company}(icl>institution).\text{def})}
\]

\[
\text{agt(\text{chairman}(icl>post).\text{present}.\text{def}, \text{John}(icl>person))}
\]

\[
\text{obj(\text{arrange}(icl>do).\text{entry}.\text{present}.\text{complete}, \text{John}(icl>person))}
\]

\[
\text{pos(\text{residence}(icl>shelter), \text{John}(icl>person))}
\]

\[
\text{obj(\text{arrange}(icl>do).\text{entry}.\text{present}.\text{complete}, \text{meeting(icl>event).\text{def})}}
\]

\[
\text{plc(\text{arrange}(icl>do).\text{entry}.\text{present}.\text{complete}, \text{residence(icl>shelter))}}
\]

In the expressions above, \text{agt} denotes the agent relation, \text{obj} the object relation, \text{plc} the place relation, \text{pos} is the possessor relation, \text{mod} is the modifier relation and \text{aoj} is the attribute-of-the-object (used to express constructs like A is B) relation. The detailed specification of the Universal Networking Language can be found at http://www.unl.iias.umass.edu/unlsys.

Our work is based on an authoritative treatise on Bengali grammar [15]. The strategies of analysis and generation of linguistic phenomena have been guided by rigorous grammatical principles.

2 EnConverter and DeConverter machines

The EnConverter (henceforth called EnCo) [16] is a language-independent parser, a multi-headed Turing machine [17] providing a framework for morphological, syntactic and semantic analysis synchronously using the UW dictionary and analysis rules. The structure of the machine is shown in the following diagram.

![Analyser Machine](Fig. 1. The EnCo machine)

The machine has two types of heads: processing heads and context heads. The processing heads (2 nos.) are called Analysis Windows (AW) and the
context heads are called *Condition Windows* (CW). The machine traverses the sentence back and forth, retrieves the relevant universal words from the lexicon and, depending on the attributes of the nodes under the AWs and those under the surrounding CWs, generates semantic relations between the UWs and/or attaches speech act attributes to them. The final output is a set of UNL expressions equivalent to a UNL graph.

The DeConverter (henceforth called the DeCo) [18] is a language-independent generator that produces sentences from UNL graphs (figure 2).

![Diagram of DeConverter]

**Fig. 2.** The DeCo machine

Like EnCo, DeCo too is a multi-headed Turing Machine. It does syntactic and morphological generation synchronously using the lexicon and the set of generation rules.

3 Rule theory

EnCo and DeCo are driven by *analysis rules* and *generation rules* respectively. These rules are *condition-action structures* that can be looked upon as programs written in a specialized language to process various complex phenomena of a natural language, both for analysis and generation. They have the following format:

```plaintext
< TYPE >
°C < PRE > "["**"]
¨<COND1>,<ACTION1>,<RELATION1>,<ROLE1>,"["**"]
°C < MID > "["**"]
¨<COND2>,<ACTION2>,<RELATION2>,<ROLE2>,"["**"]
°C < SUF > "["**"]
¨<PRIORITY ,"["
```

Characters between double quotes are the predefined delimiters of the rule. The rules mean that

- **IF**
  - under the *left processing window* there is a node satisfying <COND1> and
  - under the *right processing window* a node satisfying <COND2> attributes, and
  - there are nodes that fulfill the conditions in <PRE>, <MID> and <SUF> in the order of left, middle and right sides of processing windows respectively.

- **THEN**
  - the lexical attributes in processing windows are rewritten according to the <ACTION1>, and <ACTION2> as specified in rule, and new attributes added if necessary. (By processing window, analysis window is meant for the conversion process and generation window for the deconversion process).
  - The operations are done on the node-list depending on the <TYPE> of the rule. <RELATION1> describes the semantic relation of the node on right processing window to the node on left processing window and <RELATION2> describes the reverse [6].
  - <PRIORITY> describes the interpretation order of the rules, whose value lies between 0-255. Larger number indicates higher priority. Matching rule with the highest priority is selected for multiple matching rules.

A sequence of such rules get activated depending on the sentence situation (the conditions of the nodes under the analysis/generation windows). These are the lexico-morpho-grammatical-semantic attributes of the words under processing. For example, for a sentence like *John laughs, the animate attribute of John, the verb attribute of laugh and the adjacency of these two words under the analysis windows dictate with high probability establishing the agent (agent) relationship between the corresponding two nodes in the UNL graph.

In order to adapt the UNL engines to enconvert the Bengali sentences into the UNL interlingua and to deconvert the UNL interlingua/graph into Bengali sentences, an enconverter rule-base and a deconverter rule-base have been written. The rules within the rule-base are compliant with the corresponding UNL engines and are focused to deal with the Bengali language structure.

4 Case Structure in Bengali: *Kaaraks*

In the Indian linguistic system- descended from Sanskrit- the *case constructs* are called *kaaraks* [19]. As in the traditional understanding, they denote the relationship of the nominals with the main verb of the clause except in the *gentive case* where two nominals are related to each other. The case structure in Bengali is complex. The *kaaraks* are broadly classified into 6 types [15], each having a finer categorization into sub-types. The correspondence between the Bengali *kaaraka* system and the traditional linguistic concept of case [20] is shown by means of table 1. The *Bibhakti signa* are the case markers. An exhaustive
study of the kaaruk system with a view to analyzing Bengali into UNL has been carried out. The foundation of this work is the kaaruk theory [15]. Due to the word limitation, we exemplify the work with only the first kaaruk, viz., the kartri kaaruk.

Table 1. Case-kaaruk correspondence

<table>
<thead>
<tr>
<th>Classical Case</th>
<th>Corresponding Bengali kaaruk</th>
<th>Bibhakti signs (Case Marker)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nominative case</td>
<td>Kartrik kaaruk</td>
<td>None</td>
</tr>
<tr>
<td>Accusative case</td>
<td>Karna kaaruk</td>
<td>ke, re, err</td>
</tr>
<tr>
<td>Instrumental case</td>
<td>Karon kaaruk</td>
<td>dwaar, dya, dyja, kartrik</td>
</tr>
<tr>
<td>Dative case</td>
<td>Sampradhan kaaruk</td>
<td>janga, samutta, ke</td>
</tr>
<tr>
<td>Abitative case</td>
<td>Apanadhan kaaruk</td>
<td>theke, batte</td>
</tr>
<tr>
<td>Genitive case</td>
<td>Sambandha pat</td>
<td>r, er</td>
</tr>
<tr>
<td>Case of time and place</td>
<td>Adhikarana kaaruk</td>
<td>e, te, etc</td>
</tr>
</tbody>
</table>

4.1 Kartri kaaruk

Kartri kaaruk denotes the agent of the action stated by the verb. The kaaruk is divided into the following classes:

1. Projojak karta (প্রয়োজক কর্তা): Here the agent causes some event to take place, with an inclination towards compelling the event to happen. The morphology of the verb is exploited and the extracted knowledge has the omansate feature marked.
   Example:
   তম জোনা খেললে
tama janake khelaabe.
Tom John-to will-make-play.
Tom will make John play.

2. Nirapecchha karta (নিরুপেক্ষক কর্তা): Here there are more than one verb in the sentence with at least one Nominal (finite) verb and one Finitesimal (non-finite) verb, and the karta, i.e., agents for these verbs are different or not related. The karta associated with the non-finite verb is called the nirapecchha karta (nominal absolute in English). As there is an Finitesimal verb involved, a con or seq etc. relation is generated, also there is a possible generation of compound UW.
   Example:
   তমা খেলে জন খাবে
tama kehe janu khababe.
Tom if-eats John will-eat.
If Tom eats John will eat.

3. Karmakartribachey her karta (কর্মকর্ত্র-বচ্ছয়েকর কর্তা): Here, the actual karta is not present, and hence the karma, i.e., the object acts as the karta. As a result, there is no agt or equivalent relation generated for conceptualizing an agent of the sentence, instead, an oby relation is realized.
   Example:
   বান্ধি বলেছে
banthi bhalche.
Bucket has-filled-up.
The bucket has filled up.

4. Anukta karta (অনুক্ত কর্তা): In cases of karmata (karma bachya) and bhavata (bhaoa bachya) (which are variants of the passive voice), the karta is not emphasized on.
   Example:
   তম আজ খাবে
tom anij khabaye hay ni.
Tom of today eating not-happened.
Eating has not happened to Tom today.

5. Sabjogjata karta (সাভজোগী): Two karta are present in the same sentence, co-acting with each other to perform the action specified by the verb.
   Example:
   তমা খাবে মেটে খাবে
tama khabe mete khabe.
Tiger cow eating.
Tiger is eating with cow.

6. Bakyangsha karta (বাক্যাঙ্কূল কর্তা): Here the noun phrase as a unit acts as the karta. A noticeable fact is that this noun phrase does not have any Finitesimal (finite) verb.
   Example:
   সংস্থা জীবনমান করা কর্তিন কাজ
saptapte jibinamapan kara kathin kaaj.
Honest-way-in leading-life hard work.
Leading a life in an honest way is hard work.
(Note: Here hard work means difficult.)

7. Upabakya karta (উপাবকীয় কর্তা): Here there is a noun clause in the sentence. This noun clause conceptually acts as the karta. However, in order to retain the person information present in the verb, a different term causing agt relation has to be introduced in the sentence during enconversion. The conceptual karta actually does not get identified as a karta, instead it is identified as something different (for example, karma).
   Example:
   ভয় কথে বলে জানি
bhay kaake bale jaani.
Fear to-whom call I-know.
I know what is called fear.

8. Karta with 'e' bibhakti (তৃতীয় বিভক্তি): In spite of the presence of the e (4) bibhakti, the karta has to be identified as an agt or equivalent relation. A
salient point to note is that the *e* bibhakti can be used with all other *kaaraks* as well, so appropriate analysis has to be done to identify its functionality. Often the context of occurrence of the word and the grammatical attributes available with the word from the lexical dictionary guide in identifying the *kaarak* in case of *e* bibhakti.

**Example:**

> chaagale ghaaash khaay.

Goat grass eat.

(UNL relations generated for *kartri kaarak*: agent (*ag*), co-agent (*cag*), partner (*ptn*) etc.).

### 4.2 Other *kaaraks*

Five other *kaaraks* have been analyzed exhaustively as above.

1. **Karma kaarak** (6 subcategories): *Karma kaarak* is the person or thing on which the *kartri kaarak* executes the action stated by the sentence.  
   (UNL relations for *karma kaarak*: object (*obj*), beneficiary (*ben*), co-object (*cob*).)

2. **Karan kaarak** (5 subcategories): *Karan kaarak* is the thing or tool or method by which the *kartri kaarak* of the sentence executes the specified action.  
   (UNL relations for *karan kaarak*: instrument (*ins*), method (*met*).)

3. **Sampradaaan kaarak** (2 subcategories): *Sampradaaan kaarak* are cases where the agent (*kartri kaarak*) does something for someone or gives something to someone.  
   (UNL relations for *sampradaaan kaarak*: beneficiary (*ben*), goal (*gol*), purpose (*pur*), reason (*rsn*).)

4. **Apaadaan kaarak** (6 subcategories): This stands for the concept of sources of creation, location, position etc. All types of relations bearing the concept of *source* in some sense are eligible to come into this category.  
   (UNL relations for *apaadaan kaarak*: place-from (*plf*), time-from (*tnf*), from (*frm*), source (*src*).)

5. **Sambandha pad** (4 subcategories): If related to the next noun or pronoun, then the term having a *r* (3) or *er* (4) bibhakti is called a *sambandha pad*. *Sambandha pad* always has some *bibhakti* with it (never sunya *bibhakti*).  
   (UNL relations for *sambandha pad*: modifier (*mod*), possession (*pos*), part-of (*pof*).)

6. **Adhikaran kaarak** (8 subcategories): *Adhikaran kaarak* are the ones that describe the place, time and topic of the action performed by the sentence.  
   (UNL relations for *adhikaran kaarak*: place (*plc*), time (*tim*), place-to (*plt*), time-to (*tnf*), to (*to*), goal (*gol*), virtual-place (*sen*), objectified-place (*opl*).)

7. **Sambodhan** (3 subcategories): *Sambodhan* is the case where someone hails some other person and says something to this person. This act of hailing is captured by what is called *sambodhan*. This generates a *vocative* attribute against the called person’s appearance in the UNL graph.

Table 2 summarizes the correspondence between Bengali *kaaraks* and the UNL relations.

<table>
<thead>
<tr>
<th>Kaarak</th>
<th>Corresponding UNL Relations</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Kartri kaarak</em></td>
<td><em>ag</em>, <em>cag</em>, <em>ptn</em>, <em>aoj</em>, <em>cao</em></td>
</tr>
<tr>
<td><em>Karma kaarak</em></td>
<td><em>obj</em>, <em>ben</em>, <em>cob</em></td>
</tr>
<tr>
<td><em>Karan kaarak</em></td>
<td><em>ins</em>, <em>met</em></td>
</tr>
<tr>
<td><em>Sampradhaan kaarak</em></td>
<td><em>ben</em>, <em>gol</em>, <em>pur</em>, <em>rsn</em></td>
</tr>
<tr>
<td><em>Apaadaan kaarak</em></td>
<td><em>frm</em>, <em>src</em>, <em>plf</em>, <em>tnf</em></td>
</tr>
<tr>
<td><em>Sambandha pad</em></td>
<td><em>mod</em>, <em>pos</em>, <em>pof</em></td>
</tr>
<tr>
<td><em>Adhikaraan kaarak</em></td>
<td><em>plc</em>, <em>plt</em>, <em>tim</em>, <em>tnf</em>, <em>to</em>, <em>gol</em>, <em>scn</em>, <em>opl</em></td>
</tr>
</tbody>
</table>

The UNL relations that are not covered by the *kaaraks* in Bengali are: and (*and*), or (*or*), quantity (*qua*), proportion, rate or distribution (*per*), content (*cnt*), via (*via*), condition (*con*), sequence (*seq*), co-occurrence (*coo*), basis for expressing degree (*bas*), duration (*dur*), range: from-to (*fmt*) and manner (*man*).

### 5 Kaarak enconversion strategy

The basic idea is as follows. The non-verb primary (non-case [21]) words appearing in the sentences are one of the two types: (i) A word denoting a concept, which is a *kaarak* or *sambandha pad* or *sambodhan*, (ii) A word or *bibhakti* causing a conceptual relation to link two concepts.

The *kaaraks*, *sambandha pads* and *sambodhanas* get mapped to the UNL word concepts (UWs) after the analysis and appear in the UNL graph as nodes. The *bibhaktis* or conceptually relating words result in forming the edges of the graph which embed the logical relation between the two word-concepts. Also, there are lexical, morphological and semantic attributes in the dictionary entries of the word-concepts, which too are used to analyze the input. We illustrate the approach with an example:

> **Input to enconverter**

  > kiirtane chang baaul gaane sami maatiyo bhalo

  > (Input to enconverter)

  > kiirtan-by and baaul song-by I enchant-will

**Strategy:**

- When the *e* (8) *bibhakti* is added to and abstract noun, it becomes a candidate for the *met* relation, and hence, a +MET is added to it.
Finally, a met relation gets resolved when the node having the MET attribute and the verb becomes juxtaposed.

Salient rules:

- \{N,Na,ABS,'PLACE','CONCRETE','SCN','RSN','TIME,'BLKINSERT:+MET,+MORADD,+EADD,+BLKINSERT::\}
  \{\{[e]\},N,MOR,BLKINSERT::\};P30;
- \{N,MET,ABS,'V':met::\} \{V, METRES,;+METRES::\};P20;

UNL:

- met\(\text{enchant}(icl);doT;@entry,0;@entry,01\)
- agt(\text{enchant}(icl);doT;@entry,0;future.1;i[cl];person):0P)
- and:01(song(icl);song):0K;@entry,0;kirwana(icl);song):00\)
- mod:01(song(icl);song):0K;@entry,0;Aula(icl);song):0E

This example gives a flavor of the procedure involved. Similar procedure has been applied to all the various categories and subcategories. (Note: Kirtan and baul are two Indian blends of songs.)

6 Verification

An exhaustive verification of the system has been carried out by writing a UNL to Bengali Deconverter (i.e., generator). This uses the same lexicon as the Bengali enconversion system and a set of Bengali generation rules. The enconverted input sentences have been re-generated from the UNL graphs and manually matched for conceptual equivalence. This is a form of intra-platform verification, which verifies both the preservation of information and meaning during enconversion and its wholesome retrieval during deconversion using the appropriate rule-bases. Some examples follow. Many of the output sentences map back exactly to the same set of words and sentence structure as the input, without any divergences. However, to provide a more interesting delineation (within this short span of space) of the challenges faced, we mainly give the instances of input output divergence.

1. Projojak karta (প্রোজেক্ট কার্টা):
   Input to enco: tama janahe khelaabe
   Equivalent: তম জনকে খেলাবে
   Gloss: Tom John-to will-make-play

   Meaning: Tom will make John play.

   Output of deco: tama janahe khelaabe
   Equivalent: তম জনকে খেলাবে
   Gloss: Tom John-to will-make-play

   **Remark:** Exact match between input and output sentences.

2. Nirapekha karta (নিরাপদকতা):
   Input to enco: tama khele jana khaabe
   Equivalent: তম খেলে জন খাবে
   Gloss: Tom if-eats John will-eat

   Meaning: John will eat if Tom eats.

   Output of deco: jadi tama khaay jana khaabe
   Equivalent: জদি তম খায় জন খাবে
   Gloss: If Tom eats John will-eat

   **Remark:** This is an interesting case where the jadi (if) clause has got introduced into the output of the deconverter while it was not explicitly present in the input to the enconverter. However, it is correct as these sentences have the same sense conceptually.

3. Upabakyiya karta (উপবাক্যীয় কার্টা):
   Input to enco: bhay kaake bale jaani
   Equivalent: ভয় কাকে বলে জানি
   Gloss: Fear to-whom call I-know

   Meaning: (I) know what is called fear.

   Output of deco: aami jaani bhay kaake bale
   Equivalent: আমি জানি ভয় কাকে বলে
   Gloss: I know fear to-whom call

   **Remark:** An explicit aami (I) has been introduced in the generated sentence.

4. Bakyangsha karma (noun phrase as an object) (বাক্যাঙ্ক্ষ কর্ম):
   Input to enco: aamtaa aamtaa kathaal bhalaloabasa naa
   Equivalent: আমতা আমতা কথাল বহালোঝাসি না
   Gloss: Soft soft to-talk I-like not

   Meaning: (I) don't like to talk softly.

   Output of deco: aami bhalaloabasi naa aamtaa aamtaa kathaal balte
   Equivalent: আমি বহালোঝাসি না আমতা আমতা কথা বলে
   Gloss: I like not soft soft to-talk

   **Remark:** Conceptually these are the same, although the structures differ and order in the generated sentence is not normal in Bengali prose.

5. Karmer bipsaa (কর্মের বিপদ) (Repetition in Karma):
   Input to enco: kii kii caao bali
Equivalent: কী কে ছাড় বলি�।

Meaning: (I)/(Let me) say what (you) want.

Output of deco: aami bali tomarra kii kii caao
Equivalent: আমি বলি তোমরা কী কী ছাড়
Gloss: I say you what what want.

Remark: The input to enco has no default number information associated with the person, so the output generates (by default implementation as per the rule base) a singular number output for the first person and a plural number output for the second person. As it can be seen, an aami, which means I (first person singular number) and a tomarra, which means you (second person plural number), have been explicitly added to the output.

6. Karaner bipsaa (করানের বিস্তার) (Repetition in Karan):
Input to enco: taaraay taaraay bharaa rater aakaash
Equivalent: তারায় তারায় ভারা রাতের আকাশ
Gloss: Star-with star-with filled night’s sky

Meaning: (The) night’s sky is filled with stars.

Output of deco: rater aakaash taaraader diye bhaareche
Equivalent: রাতের আকাশ তারাদের দিয়ে ভারেচে
Gloss: Night’s sky stars-with has-filled

Remark: The structural and morphological differences in the input and output here is noticeable, although the conceptual meanings are the same in both the cases.

7. Sunnya bibhakti (no bibhakti) in karan (ব্যাপার সাবমিক বিশ্বাসে):
Input to enco: gaadhaake haajar caabuk maarleo se ghoozaa hay na
Equivalent: গাধাকে হাজার চাবুক মারলেও সে ঘোষা হয় না
Gloss: Donkey-to thousand whiplash in-spite-of-beating-with it horse become not

Meaning: In spite of thousand beatings with whiplashes a donkey does not become a horse.

Output of deco: jadi tomarra haajar caabuk diye gaadhaake maaro tabu se ghoozaa hay na
Equivalent: যদি তোমরা হাজার চাবুক দিয়ে গাধাকে মারো তবুও সে ঘোষা হয় না

Remark: These two mean the same, although the word daazDiye has come in twice in the deconverter output (to ensure the coo concept) in spite of the fact that it was present only once in the input to the enconverter.

8. Asamaapikaa kriyaaa baachak (infinite verb-related) apaadaan kaaraka (অসামাপটিক ক্রিয়াঘাতক অপাদান করুক):
Input to enco: aami marite bhiita nai
Equivalent: আমি মরে ভীত নই
Gloss: I to-die afraid not

Meaning: I am not afraid to die.

Output of deco: maraa jannya aami bhiita nai
Equivalent: মরার জন্য আমি ভীত নই
Gloss: To-die I afraid not

Remark: These, again, differ in the anusarga (jannya in the output), but the input means the same in Bengali as the output in spite of this difference in construction.

9. Saamipya suchak (proximity-denoting) adhikaran kaaraka (সামীপ্যকৃত অভিধান করুক) :
Input to enco: tama darajaay daazDiye brishti dekhche
Equivalent: তম দরাজায় দাচড়িয়ে ব্রিশ্টি দেখছে
Gloss: Tom at-door standing rainfall seeing

Meaning: Tom is seeing rainfall standing at the door.

Output of deco: tama darajaay daazDiye daazDiye brishti dekhche
Equivalent: তম দরাজায় দাচড়িয়ে দাচড়িয়ে ব্রিশ্টি দেখছে
Gloss: Tom at-door standing standing rainfall seeing

Remark: These two mean the same, although the word daazDiye has come in twice in the deconverter output (to ensure the coo concept) in spite of the fact that it was present only once in the input to the enconverter.

10. Bishyaadhikaran (topic denoting adhikaran) kaaraaka (বিশ্বাসিতকৃত করুক):
Input to enco: se taase pokta ebang futbaleo ostaad
Equivalent: সে তাতে পোক্তা এবং ফুটবলেও ওস্টাদ
Gloss: He in-cards solid and in-football expert

Meaning: He is solid in cards and expert in football.
Output of deco: futhele ostaad ebang se tause pokta
Equivalent: ফুটবলে ওসাদি এবং সে দাসে পোর্ক
Gloss: In-football expert and he in-cards solid

**Remark:** This is an instance of free-format input natural language, where the output structure has significantly varied from the input structure, in spite of having the same meaning and hence being correct.

### 7 Conclusion

Systematic analysis of the case structure forms the foundation for any natural language processing system. In this paper, we have described a system for the computational analysis of the Bengali case structure for the purpose of interlingua based MT using UNL. The complementary generator system too has been implemented, which provides the platform for intra system verification. Verification via cross system generation is being done using the Hindi generation system (also under development.) Apart from the case structure, computational analysis based on authoritative grammatical treatise, addressing complex phenomena involving verbs, adjectives and adverbs is under way.

### References