*Sādhanā* Vol. 37, Part 2, April 2012, pp. 299–318. © Indian Academy of Sciences

# Punjabi to UNL enconversion system

PARTEEK KUMAR[1,*] and R K SHARMA[2]

[1]Department of Computer Science and Engineering, Thapar University, Patiala 147001, India
[2]School of Mathematics and Computer Applications, Thapar University, Patiala 147001, India
e-mail: parteek.bhatia@thapar.edu; rksharma@thapar.edu

MS received 30 July 2010; revised 25 August 2011; accepted 19 October 2011

**Abstract.** This paper reports the work for the EnConversion of input Punjabi sentences to an interlingua representation called Universal Networking Language (UNL). The UNL system consists of two main components, namely, EnConverter (used for converting the text from a source language to UNL) and DeConverter (used for converting the text from UNL to a target language). This paper discusses the framework for designing the EnConverter for Punjabi language with a special focus on generation of UNL attributes and relations from Punjabi source text. It also describes the working of Punjabi Shallow Parser used for the processing of the input sentence, which performs the tasks of Tokenizer, Morph-analyzer, Part-of-Speech Tagger and Chunker. This paper also considers the seven phases used in the process of EnConversion of input Punjabi text to UNL representation. The paper highlights the EnConversion analysis rules used for the EnConverter and indicates its usage in the generation of UNL expressions. This paper also covers the results of implementation of Punjabi EnConverter and its evaluation on sample UNL sentences available at Spanish Language Server. The accuracy of the developed system has also been presented in this paper.

**Keywords.** DeConverter; EnConverter; machine translation; shallow parser; Universal Networking Language.



## 1. Introduction

Internet has revolutionized the way of accessing information in recent days. This information available on Internet is largely in English language and as such a good proportion of world's population is not able to access this information. Automatic translation of this information, available in the form of web pages, into other languages is a challenging task. An automatic translation tool shall certainly help in improving the popularity of Internet between masses. Several language-specific translation systems have been proposed in literature. Since these systems are based on

*For correspondence

specific source and target languages, these have their own limitations. Universal Networking Language (UNL) is an initiative to overcome the problem of language pairs in automated translation. UNL is an artificial language that is based on interlingua approach. In this approach, we need to have $n$ modules for EnConversion and other $n$ modules for DeConversion, for a given set of $n$ languages, while in the normal analysis, transfer and generation approach, we need $n^2$ modules (Uchida 1987). UNL has been developed and managed by Universal Networking Digital Language (UNDL) foundation, an international non-profit organization of Institute of Advanced Studies of United Nations University, Tokyo, Japan (Uchida *et al* 1999). This foundation has provided EnConverter and DeConverter specifications in order to convert a given Natural Language (NL) text to UNL representation and UNL representation to NL text. The foundation has also provided EnCo and DeCo tools for this purpose. Uchida & Zhu (2003) have also proposed a Universal Parser (UP) that requires a manually tagged input before NL analysis and then converting to UNL. Martins *et al* (2003) have noted that the EnCo tool provided by UNDL foundation and also this universal parser require inputs from a human expert who is seldom available and as such their performance is not quite adequate.

Dave *et al* (2001) proposed a translation system between Hindi and English languages using the tools provided by UNL foundation. Dhanabalan *et al* (2002) proposed an EnConversion tool from Tamil to UNL. They employed a specially designed parser in order to perform syntactic functional grouping. UNL based analysis and generation of Bengali case structure constructs have also been performed by Dey & Bhattacharyya (2003). Mohanty *et al* (2005) used Semantically Relatable Sequence (SRS) based approach for developing a UNL based Machine Translation (MT) system. They analysed the source language using semantic graphs and used these graphs to generate target language text. Due to non-availability of the source code of DeCo tool and its complex rule-format, Singh *et al* (2007) have proposed a DeConverter for Hindi Language.

The HERMETO system developed by Martins *et al* (2003) in the Interinstitutional Center for Computational Linguistics, Brazil converts English and Brazilian Portuguese into UNL. This system has an interface with debugging and editing facilities along with its high level syntactic and semantic grammar that make it more user friendly. Blanc (2005) proposed an EnConverter and DeConverter system for French language. The system developed by Lafourcade (2005) uses ant colony algorithm for semantic analysis and fuzzy UNL graphs for EnConversion process. Boguslavsky *et al* (2005) proposed a multi-functional linguistic processor, ETAP-3, for resolution of ambiguity in the EnConversion process. Statistical approaches have also been used to classify UNL relations. Nguyen & Ishizuka (2006) trained UNL relations classifier using statistical techniques based feature extractor. Lexicalized probabilistic parser has also been employed by Jain & Damani (2009) for English to UNL conversion. This parser has been used by them to create typed dependency tree and phase structure tree for a given English sentence. Arabic MT System based on UNL has also been developed and successfully tested. In this system, Arabic generation grammar is created for the tools for MT system (Alansary *et al* 2007; Adly & Alansary 2009).

This paper focuses on UNL based system that converts input Punjabi sentence into equivalent UNL representation. Punjabi is an Indo–Aryan language spoken by the Punjabi people in India, Pakistan, Canada and other parts of the world. The literature on MT system for Punjabi language is not that rich. Josan & Lehal (2008) have developed a Punjabi–Hindi MT system and Goyal & Lehal (2009; 2010) have developed a Hindi–Punjabi MT system using direct approach. Bhatia & Sharma (2009) analysed the role of Punjabi morphology in designing Punjabi–UNL EnConverter. They have discussed EnConversion analysis rules for resolution of relations and generation of UNL attributes. A Punjabi–Hindi machine translation system has also been

implemented by International Institute of Information Technology (IIIT), Hyderabad. In this system, traditional rules and dictionary based algorithms with statistical machine learning have been used (Anthes 2010).

In this paper, framework for designing the EnConverter for Punjabi is discussed with special focus on generation of UNL attributes and relations from Punjabi source text. The paper is divided into eight sections. Section 2 describes the structure of UNL. The analysis rules used in the UNL generation are described in section 3. The architecture of Punjabi EnConveter is discussed in section 4. Section 5 elaborates the working of Punjabi EnConverter with a sample input sentence. EnConversion of complex sentences is described in section 6. The experimentation and testing of the system is given in section 7. Section 8 concludes the work done in this paper and highlights the future directions.

## 2. UNL Structure

The process of converting a source language (natural language) expression into the UNL expression is referred to as 'EnConversion'. The process of converting UNL expressions into a target language representation is called 'DeConversion'. UNL documents are made up of Universal Words (UW); a set of relations and attributes. Information in UNL is represented sentence by sentence (Munpyo & Oliver 1999). A UW is a string of characters (in English) followed by a list of restrictions (also called as Constraint List). Relations are the building blocks of UNL sentences. One UNL construct consists of one relation and two UWs. The relations between UWs have different labels according to different roles they play (Uchida & Zhu 1993). The UNL consists of forty six relations that represent all possible associations between UWs (Uchida 2005). Attributes are also used with UWs to describe the subjectivity in the sentences (Uchida & Zhu 2001).

The English sentence '*Girls eat mangoes*' shall be represented into UNL expression as:

```
{unl}
agt(eat(icl>event).@entry.@present,
girl(icl>person).@pl)
obj(eat(icl>event).@entry.@present,
mango(icl>food).@pl)
{/unl}
```

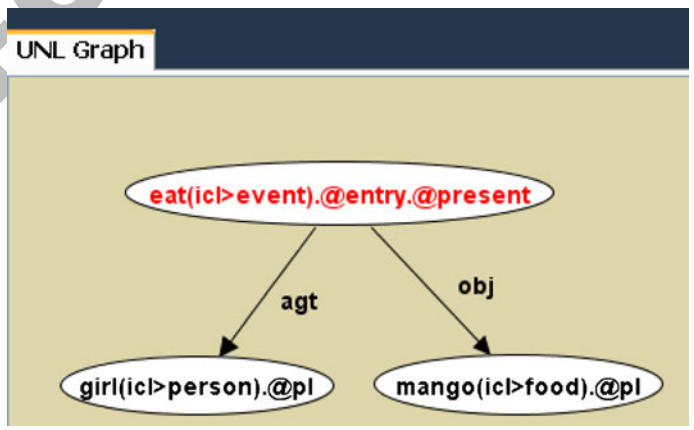


**Figure 1.** UNL graph.

UNL expressions can also be represented in the form of UNL graphs. The UNL graph of the above mentioned sentence is given in figure 1.

It may 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 @pl is UNL attribute which indicates the number information.

## 3. Working of Punjabi EnConverter

Punjabi EnConverter processes given input sentence from left to right. It uses two types of windows (Dhanabalan *et al* 2002; Bhattacharyya 2001b), namely, analysis window and condition window in the processing. The currently focused analysis windows are circumscribed by condition windows as shown in figure 2.

Here, in figure 2, 'A' indicates an analysis window, 'C' indicates a condition window, and '$n_i$' indicates an analysis node. These nodes are governed by EnConversion analysis rules. The database of these EnConversion rules is created on the basis of case markers and morphology of Punjabi language (Uchida 1987; Bhattacharyya 2001b; Dey & Bhattacharyya 2003). The rules are designed on the guidelines given in the UNDL EnConverter specifications (UNL 2000). The rule format used in designing of the system is as follows.

{COND1:ACTION1:REL1} {COND2:ACTION2:REL2}

Here,

- <COND1> indicates Condition1, it contains the lexicon attributes of left analysis window.
- <COND2> indicates Condition2, it contains the lexicon attributes of right analysis window.
- <ACTION1> and <ACTION2> are used to indicate the actions performed if the corresponding condition is true.
- The <REL1> and <REL2> fields indicate the possible relation between two analysis windows.



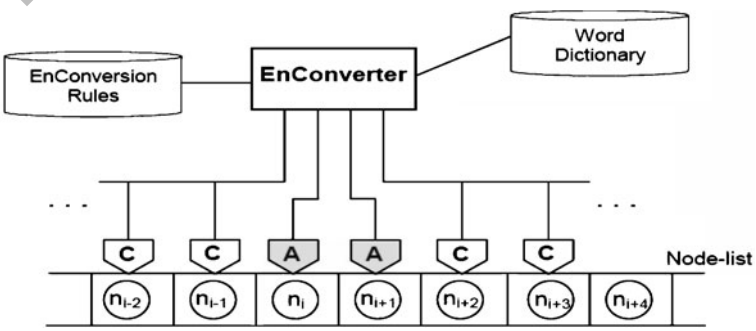


**Figure 2.** Working of Punjabi EnConverter.

## 4. Punjabi EnConverter architecture

The architecture of Punjabi EnConverter can be divided into seven phases, including an optional phase. It consists of the tasks of processing of input Punjabi Sentence by Punjabi Shallow Parser, creation of linked list of nodes on the basis of output of Shallow Parser, extraction of UWs and generation of UNL representation of the input sentence. The phases are:

(i) Parser Phase to parse the input Sentence with Punjabi Shallow Parser.
(ii) Linked List Creation Phase.
(iii) Universal Word Lookup Phase.
(iv) Case Marker Lookup Phase.
(v) Unknown Word Handling Phase.
(vi) User Interaction Phase ( This phase is optional) and
(vii) UNL Generation Phase.

Figure 3, contains the flow chart illustrating working of Punjabi EnConverter. In the next subsections, these phases are explained in brief.

### 4.1 *Parser Phase*

Punjabi EnConverter uses Punjabi Shallow Parser (developed by IIIT Hyderabad, India) for processing the input Punjabi sentence (Jain & Damani 2009). This parser performs the tasks of Tokenizer, Morph analyzer, Part-of-Speech Tagger and Chunker on the input sentence, and produces the final output by picking most appropriate Morph with Head and Vibhakti Computation. It also has the provision of using the output of each intermediate stage. It generates the output in the Shakti format (Bharati *et al* 2007).

### 4.2 *Linked list creation phase*

In this phase, Punjabi EnConverter constructs a linked list of nodes. This linked list is constructed on the basis of output generated by the Parser. It has a node for representing an individual word of the input sentence. Every node has four attributes for storing Punjabi word, Universal Word, Part-of-Speech (POS) information, and a linked list of dictionary attributes. The structure of this node is given in figure 4.

In figure 4, 'D1', 'D2' and 'D3' are the dictionary attributes of node.

### 4.3 *Universal word lookup phase*

In this phase, Punjabi-UW dictionary is used for mapping of Punjabi words to Universal Words and to retrieve the lexical-semantic information of the words. Any entry in the dictionary is stored in the format (Dave *et al* 2001; Bhattacharyya 2001a).

[HW] {ID} "UW" (ATTRIB1, ATTRIB2, ...) <FLG,FRE,PRI>;

where, HW is Head Word, ID is Identification of Head Word (this is optional), UW is Universal Word, ATTRIB is Attribute, FLG is Language Flag (e.g., E for English), FRE is Frequency of Headword and PRI is Priority of Headword.

An example of a dictionary entry is given below.

[ਕਾਰੀਗਰ] {}"artist(icl>person)" (N,MALE,ANIMT,Na) <P,0,0>;He was great artist./()

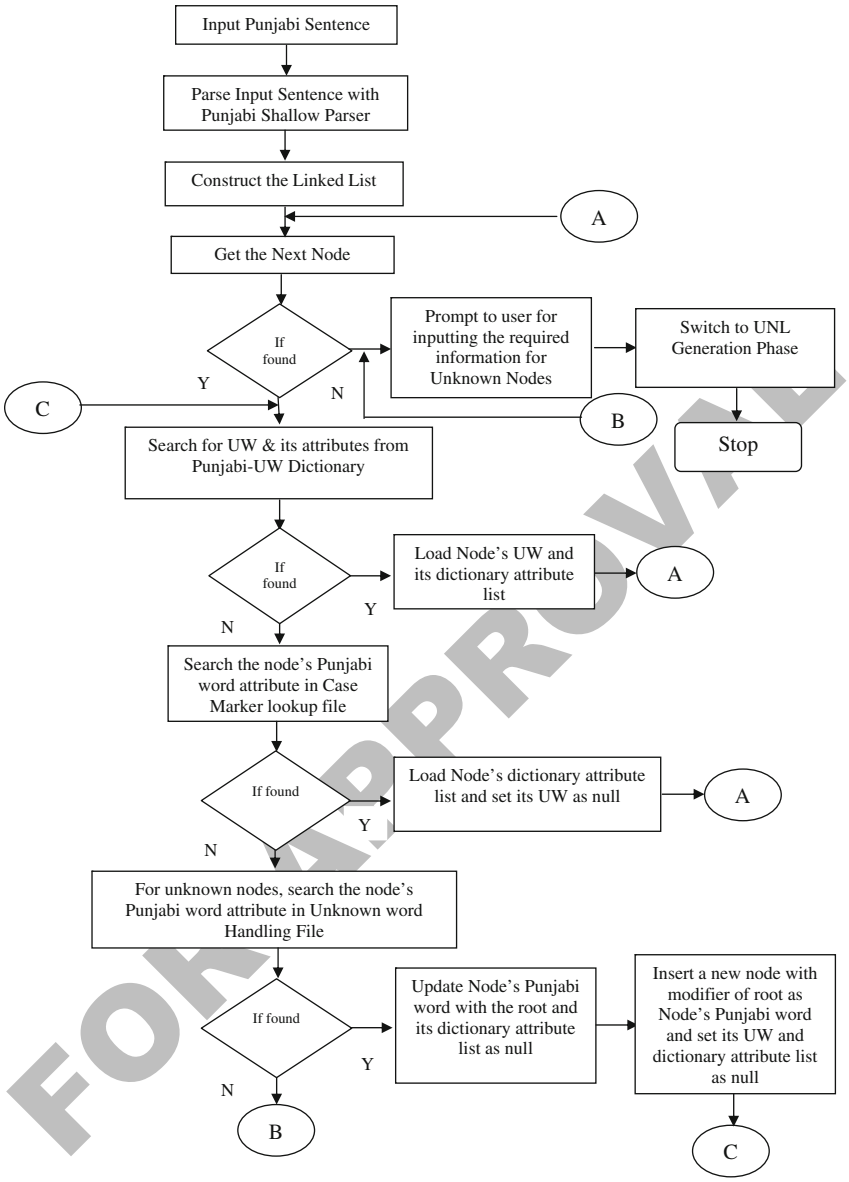




**Figure 3.** Flowchart of Punjabi EnConverter.

The attributes are used to include word category and other information such as person and number.

Exact UW is extracted from the dictionary on the basis of node's Punjabi word attribute and its grammatical category that is extracted from the Parser. After extracting the UW, the node's UW attribute is updated and linked list of dictionary attributes is extended to append the UW dictionary attributes with the attributes given by the Parser. The UW dictionary attributes provide additional details about the node such as its morphology, verbal inflexions, semantics, etc.

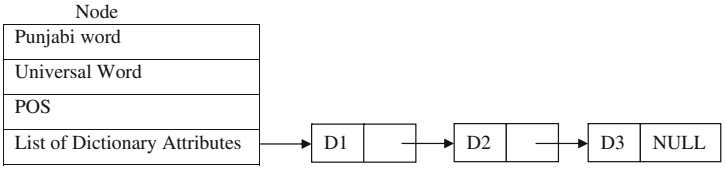


**Figure 4.** Structure of node.

Punjabi–UW dictionary may contain more than one entry for a given Punjabi word. The searching process retrieves the Universal word that matches with the node's Punjabi word and its grammatical category. For example, Punjabi word ਖੇਡ (*kheD*, English equivalent 'play') has two entries in Punjabi–UW dictionary, one as noun and other as a verb. It selects only that entry which matches with the grammatical category of the node given by the final output of Punjabi Shallow Parser. If a node is marked as unknown in the first phase by the Parser, then node's Punjabi word attribute is searched in dictionary with its grammatical category as null. In case of multiple entries of that word, the system returns the UW of first entry and thus the unknown word becomes known.

### 4.4 *Case marker lookup phase*

If a node is not found in the Punjabi–UW dictionary, then this might be a case marker of Punjabi Language having no corresponding UW. The node's Punjabi word attribute is searched in the case marker lookup file, if a word is found then the information about the case marker is added in the linked list of dictionary attributes of the node and its UW is set to null (because a case marker has no corresponding UW). This information plays an important role in resolving UNL relations in UNL generation phase.

### 4.5 *Unknown word handling phase*

As discussed earlier, there may be some words in the input sentence, which may not be resolved by the Punjabi Shallow Parser. These words are indicated as unk (unknown) by the parser. If an unknown word is found in the Universal Word Lookup Phase, then corresponding node is updated with its UW and dictionary attributes. If unknown words are not resolved in the Universal Word Lookup Phase, then these are resolved in the Case Marker Lookup Phase. If these still remain unknown, these words are processed in Unknown Word Handling Phase.

In this phase, system searches an unknown word in unknown word handling file that contains approximately one million entries of Punjabi words. It contains only those Punjabi words that are derived from some root words because all other unknown words are resolved by UW Lookup Phase or Case Marker Lookup Phase. The unknown word handling file stores Punjabi words with their corresponding root words, e.g., ਜਾਵਾਂਗਾ (*jawaNga*, English equivalent 'will go') is stored with ਜਾ (*ja*, English equivalent 'go').

If an unknown word is a derived form of a root word, then its root word is retrieved from this lookup process. The system replaces the unknown word of node's Punjabi word attribute with the root word extracted from the lookup process. The words derived from root words can be divided into two parts. One is root word and second is its suffix information. For example, Punjabi word ਜਾਵਾਂਗਾ (*jawaNga*) is derived from root word ਜਾ (*ja*) and can be divided into root ਜਾ (*ja*) and

suffix characters ਵਾਂਗਾ (*waNga*). These suffix characters play an important role in the generation of UNL attributes (Ali *et al* 2008). Thus, a new node is inserted in the linked list for these suffix characters. The Punjabi word attribute of this new node is set to these suffix characters and all other attributes are set to null. As such, in case of unknown word ਜਾਵਾਂਗਾ (*jawaNga*), node's Punjabi word attribute is set to ਜਾ (*ja*) and a new node is inserted into the linked list with its Punjabi word attribute as ਵਾਂਗਾ (*waNga*).

If a node is modified by Unknown Word Handling Phase, then it is again processed in Universal Word Lookup Phase for getting its UW otherwise it remains as unknown word as shown in figure 3.

### 4.6 *User interaction phase*

This optional phase requests the user to supply information for unknown nodes. It prompts all unknown nodes to user and requests for its UW and dictionary attributes information. If user supplies the required information, the system stores it in the node's data structure and starts the UNL generation phase; otherwise the system tries to generate the UNL with the unknown nodes.

### 4.7 *UNL generation phase*

After creation of linked list and its processing in above discussed phases, the linked list is ready for the generation of UNL representation. This phase uses approximately one thousand EnConverter analysis rules for generation of UNL representation. It invokes the following algorithm for UNL relation resolution and generation of attributes.

Algorithm 1:

(i) Process each node of linked list by considering the first node as left analysis window and the node next to this as right analysis window.
(ii) Search for the required rule from the set of EnConverter analysis rules. This depends upon the dictionary attributes of left and right analysis windows.
(iii) Modify the linked list to resolve the UNL relations and generate UNL attributes according to the fired rule. If no rule is fired, then go to step (v).
(iv) Consider first node of modified linked list as left analysis window and node next to this as right analysis window. Go to step (ii) with new analysis windows.
If the modified linked list contains a single node only, then consider that node as 'entry node' and stop further processing. It means that all the nodes are successfully processed by the system.
(v) If no rule is fired in step (ii), then shift the window to right. This effectively means that right analysis node will become the left analysis window and node next to this will become the right analysis window. Go to step (ii) with new analysis windows.

## 5. EnConversion of a simple Punjabi sentence to UNL representation: An example

In this section, working of Punjabi EnConverter is explained with the help of one example Punjabi sentence. The sentence taken for the explanation is: 'ਮੈਂ ਗੈਰੇਜ [ਵਿਚ] ਕਾਰ ਧੋਂਦਾ ਹਾਂ'. Its transliterated representation will be: '*mIN gIrej wYch kar dhUNda haN*' (Ali 2002) and English

translation of this sentence is 'I wash the car in garage'. This sentence was an input to various phases given in section 3. Processing of this sentence in these phases is explained in this section.

(i) Parser phase: The input sentence is processed by Punjabi Shallow Parser. Output of this phase is given in figure 5.

(ii) Linked list creation phase: In this phase, a Linked list with seven nodes, representing the words of input sentence, as parsed in previous phase is created. The information stored in these nodes is given below.

$Node_1$ Attributes: Punjabi Word: ਮੈਂ; UW: null; POS: PRP; Dictionary Attribute List: pn, any, sg, 1, d.
$Node_2$ Attributes: Punjabi Word: ਗੈਰੇਜ; UW: null; POS: NN; Dictionary Attribute List: unk.
$Node_3$ Attributes: Punjabi Word: ਵਿਚ; UW: null; POS: PSP; Dictionary Attribute List: psp, any, sg, d.
$Node_4$ Attributes: Punjabi Word: ਕਾਰ; UW: null; POS: NN; Dictionary Attribute List: n, f, sg, 3, d.
$Node_5$ Attributes: Punjabi Word: ਧੋ; UW: null; POS: VM; Dictionary Attribute List: v, m, sg, any.
$Node_6$ Attributes: Punjabi Word: ਦਾ; UW: null; POS: null; Dictionary Attribute List: null.
$Node_7$ Attributes: Punjabi Word: ਹਾਂ, UW: null; POS: VAUX; Dictionary Attribute List: n, f, sg, 3, d.

It has been noted that $Node_2$ corresponding to Punjabi word '[ਗੈਰੇਜ]' (*gIrej*) is classified into an unknown node, since the word is not resolved by the Punjabi Shallow Parser.

(iii) Universal word lookup phase: Now, every node of the linked list is searched into the Punjabi–UW dictionary with its Punjabi word attribute and grammatical category information extracted from the Parser. If this is found in the dictionary, node's UW attribute is

```
Final Output<Sentence id="1">
1       ((      NP
1.1     ਮੈਂ     PRP     <fs af='ਮੈਂ,pn,any,sg,1,d,,'>
        ))
2       ((      NP
2.1     ਗੈਰੇਜ   NN      <fs af='ਗੈਰੇਜ,unk,,,,,,' poslcat="NM">
2.2     ਵਿਚ     PSP     <fs af='ਵਿਚ,psp,any,sg,,d,,'>
        ))
3       ((      NP
3.1     ਕਾਰ     NN      <fs af='ਕਾਰ,n,f,sg,3,d,,'>
        ))
4       ((      VGF
4.1     ਧੋਂਦਾ   VM      <fs af='ਧੋ,v,m,sg,any,,ਦਾ,ਦਾ'>
4.2     ਹਾਂ     VAUX    <fs af='ਹਾਂ,n,f,sg,3,d,,' poslcat="NM">
        ))
</Sentence>
```

**Figure 5.** Output of Parser Phase.

loaded with UW found from the Punjabi–UW dictionary and its dictionary attribute list is also extended with the dictionary attributes extracted from the dictionary. After processing all the nodes in the Universal word lookup phase, the linked list becomes.

$Node_1$ Attributes: Punjabi Word: ਮੈਂ; UW: i(icl<person)); POS: PRP; Dictionary Attribute List: pn, any, sg, 1, d, PERPRON, ANIMT.
$Node_2$ Attributes: Punjabi Word: ਗੈਰੇਜ; UW: garage(icl>thing)); POS: NN; Dictionary Attribute List: unk, N, ANIMT, PLC.
$Node_3$ Attributes: Punjabi Word: ਵਿਚ; UW: null; POS: PSP; Dictionary Attribute List: psp, any, sg, d.
$Node_4$ Attributes: Punjabi Word: ਕਾਰ; UW: car(icl>thing)); POS: NN; Dictionary Attribute List: n, f, sg, 3, d, N.
$Node_5$ Attributes: Punjabi Word: ਧੋ; UW: wash(icl>do); POS: VM; Dictionary Attribute List: v, m, sg, any,V.
$Node_6$ Attributes: Punjabi Word: ਦਾ ; UW: null; POS: null; Dictionary Attribute List: null.
$Node_7$ Attributes: Punjabi Word: ਹਾਂ, UW: null; POS: VAUX; Dictionary Attribute List: n, f, sg, 3, d.

The unknown word '[ਗੈਰੇਜ]' (*gIrej*) in $Node_2$ becomes known in this phase as 'ਗੈਰੇਜ' (*gIrej*) has an entry in Punjabi-UW dictionary.

(iv) Case Marker Lookup Phase: Those nodes which are not found in the Punjabi–UW dictionary lookup phase are processed in this phase. It means that $Node_3$, $Node_6$ and $Node_7$ are candidates for processing in this phase. These nodes do not have any entry in the Case Marker Lookup file, and also not categorized as unknown words by the Parser. As such, the information given by the Parser is sufficient for processing.
(v) Unknown word handling phase: At this stage, there is no unknown node in the linked list. Thus, unknown word handling phase and user interaction phase are not invoked for the example sentence.
(vi) UNL Generation phase: Now the linked list is processed by the algorithm given in section 4.7, for UNL relation resolution and generation of attributes. The intermediate steps for the generation of UNL representation, for example sentence are given below. Here, the node-list is shown within '<<' and '>>' and the Analysis Windows are denoted within '[' and ']' (Bhattacharyya 2001b). Initially, the node-list will have the following structure.

$$<< \text{[ਮੈਂ]}\ \text{ਗੈਰੇਜ}\ \text{[ਵਿਚ]}\ \text{ਕਾਰ}\ \text{ਧੋ}\ \text{ਦਾ}\ \text{ਹਾਂ}\ >> \quad (1)$$

$$<<\ [mIN]\ [gIrej]\ wYch\ kar\ dhU\ da\ haN\ >>\ .$$

At this stage, no EnConversion rule is fired between left and right analysis windows. Now, the analysis windows are shifted to right. Thus, the node-list will be:

$$<< \text{ਮੈਂ}\ \text{[ਗੈਰੇਜ]}\ \text{[ਵਿਚ]}\ \text{ਕਾਰ}\ \text{ਧੋ}\ \text{ਦਾ}\ \text{ਹਾਂ}\ >>\ . \quad (2)$$

At this stage, following left composition rule is fired between left and right analysis windows.

$$+\{\text{N,PLC:} + \text{WICH:null}\}\{\text{ਵਿਚ:null:null}\}. \quad (3)$$

This rule is preceded by '+' sign, which indicates that it is a left composition rule. It results into concatenation of right node to the left node as a single composition node and the attributes of left node are inherited for further processing. The presence of '+' sign in the action part of left analysis window results into the addition of attributes. Now, WICH is added as dictionary attribute of left analysis window and the node-list becomes:

$$<< \text{ਮੈਂ [ਗੈਰੇਜ ਵਿਚ] [ਕਾਰ] ਧੋ ਦਾ ਹਾਂ} >> . \quad (4)$$

Here, no EnConversion rule is fired between left and right analysis windows, and the analysis windows are shifted to right. The node-list now becomes:

$$<< \text{ਮੈਂ ਗੈਰੇਜ ਵਿਚ [ਕਾਰ] [ਧੋ] ਦਾ ਹਾਂ} >> . \quad (5)$$

The analysis windows trigger the following rule to be fired at this level.

$$> \{\text{N,INANI,}\hat{}\text{WICH,}\hat{}\text{PLC,}\hat{}\text{SRCRES,}\hat{}\text{TOON:null:obj}\} \ \{\text{V:+OBJRES:null}\} . \quad (6)$$

As this rule is preceded by '>', it is a right modification rule. This rule is applicable when left node modifies the right node. It deletes the left node from the node-list, while the right node remains in the node-list. The presence of '^' before the dictionary attributes in the condition part of rule indicates that these attributes should not be present in corresponding node's dictionary attribute list. Here, *obj* relation is resolved between two analysis windows (due to presence of *obj* in the relation part of left analysis window) as shown below.

$$\text{obj(wash (icl>do),car (icl>thing))} . \quad (7)$$

The presence of '+' sign in the action part of right analysis window results into the addition of OBJRES attribute to right analysis window and the node-list becomes:

$$<< \text{ਮੈਂ ਗੈਰੇਜ ਵਿਚ [ਧੋ] [ਦਾ] ਹਾਂ} >> . \quad (8)$$

Now, the following left composition rule is fired between left and right analysis windows.

$$+\{\text{V:+.@sg.@male:null}\}\{[\text{ਦਾ}]\ \text{null:null}\}. \quad (9)$$

It is again a left composition rule which results into concatenation of right node to the left node as a single composition node and the attributes of left node are inherited for further processing. The presence of '+' sign in the action part of left analysis window results into the addition of attributes. Since, these attributes are preceded by '@' sign, they are concatenated to corresponding UW as UNL attributes. Now, the UW 'wash(icl>do)' is modified as 'wash(icl>do).@sg.@male' and the node-list becomes:

$$<< \text{ਮੈਂ ਗੈਰੇਜ ਵਿਚ [ਧੋ ਦਾ] [ਹਾਂ]} >> . \quad (10)$$

At this stage, following left composition rule is fired between left and right analysis windows.

$$+\{\text{V:+.@present:null}\}\{[\text{ਹਾਂ}]\text{:null:null}\}. \quad (11)$$

As discussed earlier, it is a left composition rule and results into concatenation of right node to the left node as a single composition node. After the application of this rule, the UW

'wash(icl>do).@sg.@male' becomes 'wash(icl>do).@sg.@male.@present' and the node-list becomes:

$$<< \text{[ਮੈਂ] [ਗੈਰੇਜ ਵਿਚ] ਧੋ ਦਾ ਹਾਂ} >> . \quad (12)$$

No EnConversion rule is fired between left and right analysis windows at this time. The analysis windows are shifted to right and the node-list becomes:

$$<< \text{ਮੈਂ [ਗੈਰੇਜ ਵਿਚ] [ਧੋ ਦਾ ਹਾਂ]} >> . \quad (13)$$

The rule fired between left and right analysis windows at this stage is given below.

$$> \{N,PLC,WICH:null:plc\}\ \{V:null:null\} . \quad (14)$$

It is a right modification rule that deletes the left node from the node-list, while the right node remains in the node-list. Here, *plc* relation plc(wash(icl>do).@sg.@male.@present, garage(icl>thing)) is resolved between two analysis windows.

The node-list now becomes:

$$<< \text{[ਮੈਂ] [ਧੋ ਦਾ ਹਾਂ]} >> . \quad (15)$$

Now, the rule fired between left and right analysis windows is given below.

$$> \{PERPRON,ANIMT:null:agt\}\ \{V:+AGTRES:null\} . \quad (16)$$

It is again a right modification rule. It deletes the left node from the node-list, while the right node remains in the node-list. Here, *agt* relation agt(wash(icl>do).@sg.@male.@present, i(icl<person)) is resolved between two analysis windows.

After this processing, the node-list is:

$$<< \text{[ਧੋ ਦਾ ਹਾਂ]} >> . \quad (17)$$

Now, there is a single node in the node-list. This is considered as root node and '.@entry' attribute is concatenated to its UW as given below.

$$\text{wash (icl>do) .@sg.@male.@present.@entry.} \quad (18)$$

Finally, the UNL expression generated by the Punjabi EnConverter system for input Punjabi sentence is as follows.

Punjabi sentence: ਮੈਂ ਗੈਰੇਜ ਵਿਚ ਕਾਰ ਧੋਂਦਾ ਹਾਂ..

Transliterated sentence: *mIN gIrej wYch kar dhUNda haN*.
Equivalent English sentence: I wash the car in garage.
UNL representation generated by the system:

{unl}
obj(wash(icl>do).@sg.@male.@present.@entry,car(icl>thing))
plc(wash(icl>do).@sg.@male.@present.@entry,garage(icl>thing))
agt(wash(icl>do.@sg.@male.@present.@entry,I(icl<person))
{/unl}.

## 6. EnConversion of complex sentences

A complex sentence is the sentence in which we have one main clause and one (or more) subordinate clause(s). In this paper, we have dealt with the complex sentences involving adverb clause with one main and one subordinate clause. A critical issue in the analysis of complex sentences is to find the words acting as clause delimiters. The next issue is to relate the clause to the correct word in the sentence using appropriate relation, e.g., an adverb clause should be related with the verb of another clause while an adjective clause to the noun.

### 6.1 *UNL Representation of complex sentences*

Complex sentences are represented in UNL with the help of compound UWs. A compound UW is a set of binary relations that are grouped together as a single unit. In UNL, compound UWs are represented by compound UW-IDs, which start with ':' followed by two numbers (each between 0 and 9), for example ':01'. This concept of compound universal words is illustrated below.

Let us consider the sentence, 'Ram, who lives in Delhi, eats rice'. Its UNL representation is:

```
{unl}
agt(eat(agt>person,obj>food),:01)
obj(eat(agt>person,obj>food),rice(icl>food))
agt:01(live(agt>person), Ram)
plc:01(live(agt>person), Delhi(icl>person))
{/unl}
```

Here, ':01' indicates the compound UW. Figure 6, consists of UNL graph of this example representation.

### 6.2 *Role of adverb clause in complex sentences*

Adverb clause is a subordinate clause which acts as adverb in a sentence. It may modify some verb, adjective or adverb in the main clause. In order to resolve UNL relations for adverb clauses, these are classified as adverb clause for time, for condition, for place and for manner (Giri 2000).

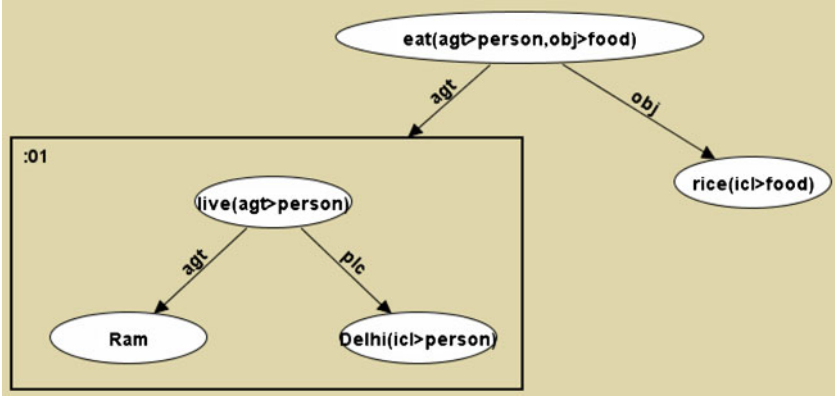


**Figure 6.** UNL Graph with compound UW.

All the adverb clauses have a common dictionary attribute ADV. The types of adverb clause along with attributes used and corresponding Punjabi word to represent these types, is depicted in table 1.

### 6.3 *EnConversion of adverb clause for time sentences*

The adverb clause for time is resolved between the verb of the main clause and the verb of the subordinate clause using the attributes ADV and TCL. The adverb clause for time is represented by Punjabi words ਜਦ (*jd*) and ਤਦ (*td*). The word ਜਦ (*jd*) indicates the beginning of the time condition, so it is represented by ADV, TCLB attributes, while word ਤਦ (*td*) indicates the action as a result of that time and is represented by ADV, TCL attributes (Giri 2000). The handling of adverb clause of time is illustrated with the Punjabi sentence, "ਜਦ ਮੈਂ ਅਤੇ ਸੀਤਾ ਟੋਕੀਓ ਜਾਵਾਂਗੇ ਤਦ ਅਸੀਂ ਵੱਡਾ ਮੀਨਾਰ ਅਤੇ ਇਮਾਰਤ ਦੇਖਾਂਗੇ". Its transliterated representation will be '*jd mIN Ote sYIta TUkYIU jawaNge td OsYIN wWDa mYInar Ote aYmart dekhaNge*', and English translation of this sentence is 'When I and Sita will go to Tokyo, then we will see big tower and building'.

We have used following left composition rule to add ADV, TCLB to the main verb of the sentence.

$$-\{\text{ADV,TCLB:null:null}\}\ \{\text{PERPRON:} + \text{ADV}, + \text{TCLB:null}\}\,. \tag{19}$$

This rule is preceded by '$-$' sign, which indicates that it is a right composition rule. It results into concatenation of left node to the right node as a single composition node and the attributes of right node are inherited for further processing. The presence of '$+$' sign in the action part of right analysis window results into the addition of attributes to the corresponding node. This rule concatenates the adverb of time, recognized by the presence of the attribute ADV and TCLB with the pronoun on the right. After the application of this rule, personal pronoun gets the attributes ADV, TCLB in addition of its existing attributes. When other relations are resolved with this personal pronoun, it retains these attributes. It means that when *agt* relation is resolved between ਮੈਂ (*mIN,* 'i') and ਜਾ (*ja,* 'go'), the main verb inherits all the attributes of pronoun besides retaining its existing attributes.

Similarly, adverb ਤਦ (*td*) is concatenated with the personal pronoun ਅਸੀਂ (*OsYIN*) by using following composition rule.

$$-\{\text{ADV,TCL:null:null}\}\ \{\text{PERPRON:} + \text{ADV}, + \text{TCL:null}\}\,. \tag{20}$$

As discussed earlier, when other relations are resolved with this personal pronoun it will retain these attributes. It means that when *agt* relation is resolved between ਅਸੀਂ (*OsYIN,* 'we') and ਦੇਖ (*dekh,* 'see') the verb inherits all the attributes of pronoun besides retaining its existing attributes.

**Table 1.** Types of adverb clause.

| Type of clause | Attributes used | Punjabi words used to represent adverb clause |
|---|---|---|
| Time adverb clause | ADV, TCLB, TCL | ਜਦ (*jd*) and ਤਦ (*td*) |
| Conditional adverb clause | ADV, TCLB, TCL | ਜੇਕਰ/ਜੇ (*jekr/je*) and ਤਾਂ/ਤੇ (*taN/te*) |
| Place adverb clause | ADV, PLCB, PLC | ਜਿੱਥੇ (*jYWthe*) and ਉੱਥੇ (*auWthe*) |
| Manner adverb clause | ADV, MCLB, MCL | ਜਿੱਦਾਂ (*jYWdaN*) and ਓਦਾਂ (*aUdaN*) |

After all relations of main and subordinate clause have been resolved, there remains only the verb of main clause and the verb of subordinate clause. The attributes of the verb of main clause are V, ADV, TCLB while the attributes of the verb of subordinate clause are V, ADV,TCL. Here, *tim* relation is resolved by the application of the following rule.

$$>\{\text{V, ADV, TCLB: +.@entry:null}\}\ \{\text{V, ADV, TCL:null:tim}\}. \tag{21}$$

It is a right modification rule, which results into deletion of left node from the node-list, while the right node remains in the node-list. Here, *tim* relation is resolved between two analysis windows with left analysis window (main clause) acting as parent of the relation.

The UNL representation generated by Punjabi EnConverter for the above-mentioned example sentence is given below.

```
{unl}
and(sita(icl<person),I(icl<person))
plt(go(icl>do).@entry.@future,Tokyo(icl>place))
agt(go(icl>do).@entry.@future,I(icl<person))
mod:01(tower(icl>building),large(aoj>thing))
and:01(tower(icl>building),building(icl>place))
obj:01(see(icl<event).@future.@pl.@entry,tower(icl>building))
agt:01(see(icl<event).@future.@pl.@entry,we)
tim(go(icl>do).@entry.@future,:01)
{/unl}
```

Again, ':01' indicates that it is a complex node. The UNL graph of the example sentence is given in figure 7.



## 6.4 *EnConversion of conditional adverb clause sentences*

The adverb clause for condition is resolved between the verb of the main clause and the verb of the subordinate clause using the attributes ADV and CCL. The adverb clause of condition is represented by Punjabi words ਜੇਕਰ (*jekr*) and ਤਾਂ (*taN*). The word ਜੇਕਰ (*jekr*) indicates beginning

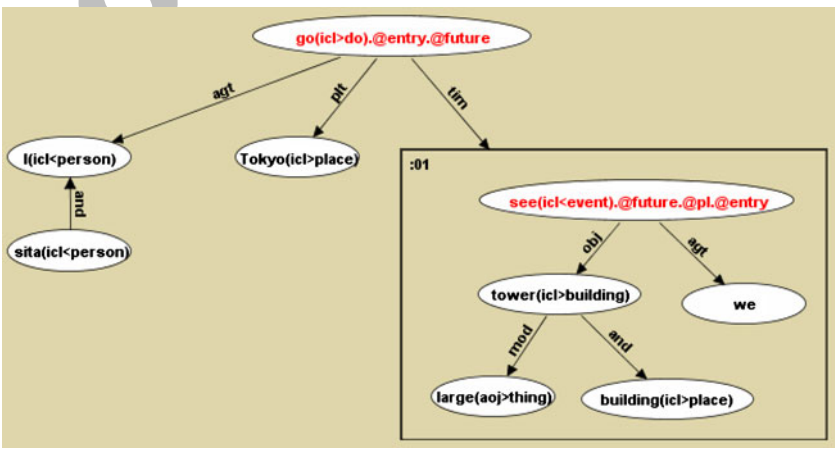


**Figure 7.** UNL Graph for example sentence with adverb clause for time.

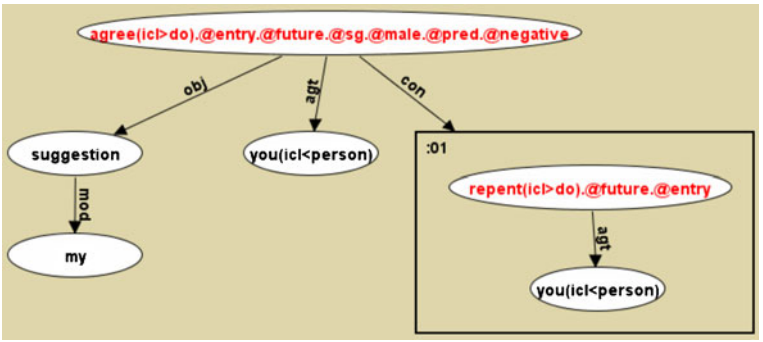


**Figure 8.** UNL Graph for example sentence for conditional adverb clause.

of the condition, so it is represented by ADV, CCLB attributes while word ਤਾਂ (*taN*) indicates the action as a result of condition and is represented by ADV, CCL attributes. The adverb clause for condition is handled in the similar manner as adverb clause for time (Giri 2000). Some of the important rules fired in the process of EnConversion of conditional adverb clause sentences are as follows.

$$-\{\text{ADV,CCLB:null:null}\}\ \{\text{PRON:+ADV,+CCLB:null}\}\ . \tag{22}$$

$$-\{\text{ADV,CCL:null:null}\}\ \{\text{PRON:+ADV,+CCL:null}\}\ . \tag{23}$$

$$>\{\text{V, ADV, CCLB: +.@entry:null}\}\ \{\text{V, ADV, CCL:null:con}\}\ . \tag{24}$$

These rules help in obtaining the UNL representation of complex Punjabi sentences for adverb clause for condition. The UNL representation generated by the system for Punjabi sentence ‘ਜੇਕਰ ਤੂੰ ਮੇਰਾ ਸੁਝਾਅ ਨਹੀਂ ਮੰਨਿਆ ਤਾਂ ਤੂੰ ਪਛਤਾਵੇਂਗਾ’, having transliterated representation as ‘*jekr tuUM mera sujhaO nhYIN mMnYA taN tuUM pchhtaweNga*’, and English translation as ‘If you will not follow my suggestion, then you will repent’ is given below.

```
{unl}
mod(suggestion,my)
obj(agree(icl>do).@entry.@future.@sg.@male.@pred.@negative,suggestion)
agt(agree(icl>do).@entry.@future.@sg.@male.@pred.@negative,you(icl<person))
agt:01(repent(icl>do).@future.@entry,you(icl<person))
con(agree(icl>do).@entry.@negative.@pred.@future.@sg.@male,:01)
{/unl}
```

The UNL graph for this example sentence is given in figure 8.

The EnConverter handles the sentences, with adverb clause for place and sentences with adverb clause for manner, successfully using the similar approach.

## 7. Evaluation of the Punjabi EnConverter

We have evaluated the performance of proposed EnConverter vis-a-vis the performance of a gold standard EnConverter. The EnConverter available at Spanish UNL language server with

**Table 2.** Results of evaluation of Punjabi EnConveter for each UNL relation.

| Sr. No. | UNL Relation | Number of sentences tested | Number of sentences with correct UNL representation generated by proposed EnConverter | Percentage of accuracy of proposed EnConverter |
|---|---|---|---|---|
| 1 | agt | 56 | 56 | 100.00 |
| 2 | and | 3 | 2 | 66.67 |
| 3 | aoj | 14 | 13 | 92.86 |
| 4 | bas | 3 | 3 | 100.00 |
| 5 | ben | 7 | 6 | 85.71 |
| 6 | cag | 2 | 2 | 100.00 |
| 7 | cao | 1 | 1 | 100.00 |
| 8 | cnt | 2 | 2 | 100.00 |
| 9 | cob | 3 | 2 | 66.67 |
| 10 | con | 1 | 1 | 100.00 |
| 11 | coo | 2 | 2 | 100.00 |
| 12 | dur | 3 | 3 | 100.00 |
| 13 | fmt | 3 | 3 | 100.00 |
| 14 | frm | 1 | 1 | 100.00 |
| 15 | gol | 4 | 4 | 100.00 |
| 16 | ins | 2 | 2 | 100.00 |
| 17 | man | 10 | 9 | 90.00 |
| 18 | met | 3 | 2 | 66.67 |
| 19 | mod | 12 | 10 | 83.33 |
| 20 | nam | 2 | 2 | 100.00 |
| 21 | obj | 48 | 45 | 93.75 |
| 22 | opl | 2 | 2 | 100.00 |
| 23 | or | 2 | 2 | 100.00 |
| 24 | per | 2 | 2 | 100.00 |
| 25 | plc | 3 | 3 | 100.00 |
| 26 | plf | 3 | 3 | 100.00 |
| 27 | plt | 2 | 2 | 100.00 |
| 28 | pof | 1 | 1 | 100.00 |
| 29 | pos | 8 | 7 | 87.50 |
| 30 | ptn | 3 | 3 | 100.00 |
| 31 | pur | 3 | 2 | 66.67 |
| 32 | qua | 7 | 6 | 85.71 |
| 33 | rsn | 2 | 2 | 100.00 |
| 34 | scn | 2 | 2 | 100.00 |
| 35 | seq | 1 | 1 | 100.00 |
| 36 | src | 3 | 3 | 100.00 |
| 37 | tim | 1 | 1 | 100.00 |
| 38 | tmf | 2 | 2 | 100.00 |
| 39 | tmt | 2 | 2 | 100.00 |
| 40 | to | 1 | 1 | 100.00 |
| 41 | via | 2 | 2 | 100.00 |

URL http://opera.dia.fi.upm.es/cle/demo_interactiva/interactive_main_page.htm has been taken as gold standard in the present work.

We have manually translated the English sentences given at Spanish language server into equivalent Punjabi sentences and then inputted those equivalent Punjabi sentences to the proposed Punjabi–UNL EnConverter system. We have compared the UNL expressions generated by our system with the UNL expressions generated by Spanish UNL language server. We consider the two UNL representations matched with each other if the relations, including associated UWs, present in the representation are same (Jain & Damani 2009). The proposed EnConversion system has been tested for ninety one Punjabi sentences. It has been seen that the system successfully handles the resolution of UNL relations and generation of attributes for these sentences. The quality of Punjabi EnConverter is measured by the number of entries that match between the UNL generated by gold standard system and the one generated by the proposed system. For the sample ninety one sentences, the two systems match for eighty seven sentences. The results of this evaluation on the basis of each UNL relation are given in table 2.

It has been noted that out of forty six UNL relations, forty one relations are used in gold standard UNL sentences. The proposed system has 95% of accuracy. In some cases, the proposed system resolves incorrect relation. The main reason for this failure is the incorrect parsing of the input sentence by Punjabi Shallow Parser. Another reason is the absence of semantic information for a large number of verbs, as this information some time plays an important role in resolving correct UNL relation.



## 8. Conclusion and future scope

In this paper, a Punjabi EnConverter has been discussed, developed, implemented and tested. Punjabi EnConverter uses EnConversion analysis rules for UNL relation resolution and generation of attributes from the input Punjabi sentence. We have here developed approximately one thousand EnConversion analysis rules for the development of Punjabi EnConverter. This proposed EnConverter has been tested for its performance vis-a-vis an EnConverter available in public domain on the Spanish Language Server. The results of testing are encouraging and the outputs of our system and the Spanish Language Server are in good agreement.

The system still needs many improvements like handling parser errors, improving rule base, compiling verb properties, etc. We are also working on implementing the word sense disambiguation module in the proposed Punjabi EnConverter.

## References

Adly N, Alansary S 2009 Evaluation of Arabic machine translation system based on the Universal Networking Language. *Natural Language Processing and Information Systems, Lect. Notes in Comp. Sci. (LNCS)* 5723:243–257

Alansary S, Nagi M, Adly N 2007 A Semantic-based approach for multilingual translation of massive documents. *Proc. 7th Int. Symposium on Natural Language Processing*, Chonburi, Thailand, 317–323

Ali M 2002 Transliteration Tables. *Proc. 7th issue of Technology Development for Indian Languages (TDIL) News Letter*: 22–23

Ali Y, Das J K, Abdullah S M, Nurannabi A M 2008 Morphological analysis of Bangla words for Universal Networking Language. *Proc. 3rd Int. Conf. on Digital Information Management*, London, England, 532–537

Anthes G 2010 Automated translation of Indian languages. *Comm. ACM* 53(1):24–26

Bharati A, Sangal R, Sharma D 2007 *SSF: Shakti Standard Format Guide*. Language Technologies Research Centre, International Institute of Information Technology, Hyderabad, India: 1–25

Bhatia P, Sharma R K 2009 Role of Punjabi morphology in designing Punjabi-UNL EnConverter. *Proc. Int. Conf. on Advances in Computing, Communication and Control*, Mumbai, India, 562–566

Bhattacharyya P 2001a Multilingual information processing through Universal Networking Language. *Indo UK Workshop on Language Engineering for South Asian Languages*, Mumbai, India, 1–10

Bhattacharyya P 2001b Knowledge extraction from Hindi text. *J. Inst. Electron. Telecommun. Eng.* 18(4): 1–12

Blanc E 2005 About and around the French EnConverter and the French DeConverter. *Universal Network Language: Advances in Theory and Applications*, J Cardeñosa, A Gelbukh, E Tovar, Ed(s) México, Research on Computing Science, 157–166

Boguslavsky I M, Iomdin L, Sizov V G 2005 Interactive EnConversion by means of the ETAP-3 system. *Universal Network Language: Advances in Theory and Applications*, J Cardeñosa, A Gelbukh, E Tovar, Ed(s) México, Research on Computing Science, 230–240

Dave S, Parikh J, Bhattacharyya P 2001 Interlingua based English Hindi machine translation and language divergence. *J. Mach. Trans. (JMT)* 16(4):251–304

Dey K, Bhattacharyya P 2003 Universal networking language based analysis and generation of Bengali case structure constructs. *Proc. Int. Conf. on the Convergence of Knowledge, Culture, Language and Information Technologies*, Egypt, 1–6

Dhanabalan T, Saravanan K, Geetha T V 2002 Tamil to UNL EnConverter. *Proc. Int. Conf. on Universal Knowledge and Language*, Goa, India, 1–16

Giri L 2000 *Semantic Net like knowledge structure generation from natural languages*. M. Tech. Thesis, IIT Mumbai, India

Goyal V, Lehal G S 2009 Evaluation of Hindi to Punjabi machine translation system. *Int. J. Comput. Sci.* 4(1):36–39

Goyal V, Lehal G S 2010 Web based Hindi to Punjabi machine translation system. *J. Emer. Tech. in Web Intel.* 2(2):148–151

Jain M, Damani O P 2009 English to UNL (Interlingua) EnConversion. *Proc. 2nd Conference on Language and Technology*, Lahore, Pakistan: 1–8

Josan G S, Lehal G S 2008 A Punjabi to Hindi machine translation system. *Proc. Coling: Companion volume: Posters and Demonstrations*, Manchester, UK, 149–152

Lafourcade M 2005 Semantic analysis through ant algorithms, conceptual vectors and fuzzy UNL graphs. *Universal Network Language: Advances in Theory and Applications*, J Cardeñosa, A Gelbukh, E Tovar, Ed(s) México, Research on Computing Science, 125–137

Martins R, Hasegawa R, Graças M and Nunes V 2003 HERMETO: A NL-UNL EnConverting Environment. *Proc. Int. Conf. on the Convergence of Knowledge, Culture, Language and Information Technologies*, Alexandria, Egypt, 1–4

Mohanty R, Dutta A, Bhattacharyya P 2005 Semantically relatable sets: building blocks for representing semantics. *Proc. 10th Machine Translation Summit*, Phuket, 1–8

Munpyo H, Oliver S 1999 Overcoming the language barriers in the web: The UNL-approach. *GLDV*, Deutschland, 253–262

Nguyen P T, Ishizuka M 2006 A statistical approach for Universal Networking Language-based relation extraction. *Proc. Int. Conf. on Research, Innovation and Vision for the Future*, Ho Chi Minh City, Vietnam, 153–160

Singh S, Dalal M, Vachani V, Bhattacharyya P, Damani O P 2007 Hindi generation from interlingua. *Proc. Machine Translation Summit*, Copenhagen, 1–8

Uchida H 1987 ATLAS: Fujitsu machine translation system. *Proc. Machine Translation Summit*, Japan, 129–134

Uchida H 2005 *Universal Networking Language (UNL): Specifications version 2005*, UNDL Foundation

Uchida H, Zhu M 1993 Interlingua for multilingual machine translation. *Proc. 4th Machine Translation Summit*, Japan, 157–169

Uchida H, Zhu M 2001 The Universal Networking Language beyond machine translation. *Proc. Int. Symposium on Language in Cyberspace*, Seoul, Korea, 1–15

Uchida H, Zhu M 2003 Universal Parser. UNL Center UNDL Foundation

Uchida H, Zhu M, Senta T D 1999 *The UNL, A Gift for a Millennium*. Tokyo, Japan, Institute of Advanced Studies, The United Nations University: 1–62

Universal Networking Language (UNL) Center 2000 *EnConverter Specifications*, UNDL Foundation: 1–33