# Interlingua-based English–Hindi Machine Translation and Language Divergence

SHACHI DAVE, JIGNASHU PARIKH and PUSHPAK BHATTACHARYYA
*Department of Computer Science and Engineering, Indian Institute of Technology, Mumbai, India*
*E-mail: pb@cse.iitb.ac.in, jignashu@csa.iisc.ernet.in, sdave@usc.edu*

**Abstract.** Interlingua and transfer based approaches to machine translation have long been in use in competing and complementary ways. The former proves economical in situations where translation among multiple languages is involved, and can be used as a knowledge-representation scheme. But given a particular interlingua, its adoption depends on its ability (a) to capture the knowledge in texts precisely and accurately and (b) to handle cross-language divergences. This paper studies the language divergence between English and Hindi and its implication to machine translation between these languages using the Universal Networking Language (UNL). UNL has been introduced by the United Nations University (UNU), Tokyo, to facilitate the transfer and exchange of information over the internet. The representation works at the level of single sentences and defines a semantic net-like structure in which nodes are word concepts and arcs are semantic relations between these concepts. The language divergences between Hindi, an Indo-European language, and English can be considered as representing the divergences between the SOV and SVO classes of languages. The work presented here is the only one to our knowledge that describes language divergence phenomena in the framework of computational linguistics through a South Asian language.

**Keywords**: interlingua, language divergence, analysis, generation, Universal Networking Language, Hindi

## 1. Introduction

The “digital divide” among people arises not only from the infrastructural factors like personal computers and high-speed networks, but also from the

language barrier. This barrier appears whenever the language in which information is presented is not known to the receiver of that information. The World Wide Web contents are mostly in English and cannot be accessed without some proficiency in this language. This is true for other languages too. The Universal Networking Language (UNL) has been proposed by the United Nations University (UNU) for overcoming the language barrier. However, a particular interlingua can be adopted only if it can capture the knowledge present in natural-language documents precisely and accurately. Also it should have the ability to handle cross-language divergences. Our work investigates the efficacy of the UNL as an interlingua in the context of the language divergences between Hindi and English. The language divergence between these two languages can be considered representative of the divergences between the SOV and SVO classes of languages.

Researchers have long been investigating the interlingua approach to MT and some of them have considered the widely used transfer approach as the better alternative (Vauquois and Boitet, 1985; Boitet, 1988; Arnold and Sadler, 1990). In the transfer approach, some amount of text analysis is done in the context of the source language and then some processing is carried out on the translated text in the context of the target language. But the bulk of the work is done on the comparative information on the specific pair of languages. The arguments in favour of the transfer approach to MT are (a) the sheer difficulty of designing a single interlingua that can be all things to all languages and (b) the fact that translation is, by its very nature, an exercise in comparative linguistics. The Eurotra system (Arnold and des Tombes, 1987; King and Perschke, 1987; Perschke, 1989; Schütz et al., 1991) in which groups from all the countries of the European Union participated, is based on the transfer approach. So is the Verbmobil system (Wahlster, 1993) sponsored by the German Federal Ministry for Research and Technology.

However, since the late 1980s, the interlingua approach has gained momentum with commercial interlingua-based MT systems being implemented. PIVOT of NEC (Muraki, 1987; Okumura et al., 1991), ATLAS II of Fujitsu (Uchida, 1989), Rosetta of Phillips (Landsbergen, 1987) and BSO (Witkam, 1988; Schubert, 1988) in the Netherlands are the examples in point. In the last mentioned, the interlingua is not a specially designed language, but Esperanto. It is more economical to use an interlingua if translation among multiple languages is required. Only $2n$ converters will have to be written, as opposed to $n\ (n–1)$ converters in the transfer approach, where $n$ is the number of languages involved.

The interlingua approach can be broadly classified into (a) primitive-based and (b) deeper knowledge representation-based. Examples of the former include Schank's (1972, 1973, 1975; Schank and Abelson, 1977; Lytinen and Schank, 1982) use of Conceptual Dependency (CD), the UNITRAN system (Dorr, 1992, 1993) using Lexical Conceptual Structure (LCS) and Wilk's (1972) system, while CETA (Vauquois, 1975), KBMT (Carbonell and Tomita, 1987; (Nirenburg et al., 1992), TRANSLATOR (Nirenburg, et al., 1987), PIVOT (Muraki, 1987) and Atlas (Uchida, 1989) are the examples of the latter. The UNL falls into the latter category.

Dorr (1993) describes how language divergences can be handled using the LCS as the interlingua in the UNITRAN system. The argument is that it is the complex divergences that necessitate the use of an interlingua representation. This is because of the fact that such a representation allows surface syntactic distinctions to be represented at a level that is independent of the underlying meanings of the source and target sentences. Factoring out these distinctions allows cross-linguistic generalisations to be captured at the level of the lexical-semantic structure.

The work presented here is the only one to our knowledge that describes language divergences between Hindi and English in a formal way from the point of view of computational linguistics. However, several studies by the linguistic community bring out the differences between the western and Indian languages (Bholanath, 1987; Gopinathan, 1993). These are presented in Section 5.

Many systems have been developed in India for translation to and from Indian languages. The Anusaaraka system, based on the Paninian Grammar (Bharati et al., 1995), renders text from one Indian language into another. It analyses the source-language text and presents the information in the target language retaining a flavour of the source language. The grammaticality constraint is relaxed and a special-purpose notation is devised. The aim of this system is to allow language access and not MT. IIT Kanpur is involved in designing translation support systems called Anglabharati and Anubharati. These are for MT between English and Indian languages and also among Indian languages (Bhandari, 2002). The approach is based on the word-expert model utilizing the karaka theory, a pattern-directed rule base and a hybrid example base. In MaTra (Rao et al., 2000), a human-aided translation system for English to Hindi, the focus is on the innovative use of the human–computer synergy. The system breaks an English

sentence into chunks and displays it using an intuitive browser-like representation that the user can verify and correct. The Hindi sentence is generated after the system has resolved the ambiguities and the lexical absence of words with the help of the user.

We now give a brief introduction to the UNL. It is an interlingua that has been proposed by the UNU to access, transfer and process information on the internet in the natural languages of the world. UNL represents information sentence by sentence. Each sentence is converted into a hypergraph having concepts as nodes and relations as directed arcs. Concepts are called **Universal Words** (UWs). The knowledge within a document is expressed in three dimensions:

a. Word knowledge is represented by UWs which are language independent. These UWs have restrictions that describe the sense of the word. For example, `drink(icl>liquor)`denotes the noun *liquor*. The `icl` notation indicates inclusion and forms an "is-a" structure as in semantic nets (Woods, 1985). The UWs are picked up from the lexicon during the analysis into or generation from the UNL expressions. The entries in the lexicon have syntactic and semantic attributes. The former depend on the language word while the latter are obtained from the language-independent ontology.

b. Conceptual knowledge is captured by relating UWs through the standard set of **Relation Labels** (RLs) (UNL, 1998). For example, the sentence in (1a) is described in UNL as in (1b).

(1) a. Humans affect the environment.

b.
```
agt(affect(icl>do).@present.@entry:01,
    human(icl>animal).@pl:I3)
obj(affect(icl>do).@present.@entry:01,
    environment(icl>abstract thing).@pl:I3)
```

`agt` means agent and `obj` object. `affect(icl>do)`, `human(icl>animal)` and `environment(icl>abstract thing)` are the UWs denoting concepts.

c. Speaker's view, aspect, time of the event, *etc.* are captured by **Attribute Labels**. For instance, in (1), the attribute `@entry` denotes the main predicate of the sentence, `@present` the present tense and `@pl` the plural number.

The total number of relations in the UNL is currently 41. All these relations are binary and are expressed as $rel(UW_1,UW_2)$, where $UW_1$ and $UW_2$ are UWs or compound UW labels. A compound UW is a set of binary relations grouped together and regarded as one UW. UWs are made up of a character string (usually an English-language word) followed by a list of

restrictions. When used in UNL expressions, a list of attributes and often an instance ID follow these Uws.

We explain the entities in the BNF rule (2). The Head Word is an English word or a phrase or a sentence that is interpreted as a label for a set of concepts. This is also called a **basic UW** (which is without restrictions). For example, the basic UW *drink*, with no constraint list, denotes the concepts of 'putting liquids in the mouth', 'liquids that are put in the mouth', 'liquids with alcohol', 'absorb' and so on.

(2) <UW>::=<head word>[<constraint list>][: <UW ID>][. <attribute list>]

The constraint list restricts the interpretation of a UW to a specific concept. For example, the restricted UW `drink(icl>do,obj>liquid)` denotes the concept of 'putting liquids into the mouth'. Words from different languages are linked to these disambiguated UWs and are assigned syntactic and semantic attributes. This forms the core of the lexicon building activity.

The UW ID is an integer, preceded by a colon, which indicates the occurrence of two different instances of the same concept. The constraint list can be followed by a list of attributes, which provides information about how the concept is being used in a particular sentence. A UNL expression can also be expressed as a UNL graph. For example, the UNL expressions for the sentence in (3) are shown in the top half of Figure 1, and the UNL graph for the sentence is given in the bottom half.

(3) John, who is the chairman of the company, has arranged a meeting at his residence.

In Figure 1, `plc` denotes the place relation, `pos` is the possessor relation, `mod` is the modifier relation and `aoj` is the attribute-of-the-object relation (used to express constructs like *A is B*).

The **international project on the UNL** involves researchers from 14 countries of the world and includes 12 languages. For almost all the languages, the generator from the UNL expressions is quite mature. For the process of analysis into the UNL form, classical and difficult problems like ambiguity and anaphora are being addressed. All the research groups have to use the same repository of the universal words, which is maintained by the UNDL foundation at Geneva and the UNU at Tokyo. When a new UW is coined by a research team it is placed in the UW repository at the UNU site. The restrictions are drawn from the knowledge base, which again is

maintained by the UNU. Individual teams have the responsibility of creating their local language servers, which provide the services with respect to the analysis into and generation from UNL expressions.

This paper is organized as follows. The conceptual foundations, dealing with the formalisation of the UNL system and the universality of the lexicon, are given in Section 2. Section 3 describes the use of lexical resources in semi-automatically constructing a semantically rich dictionary. Section 4 explains the working of the language-independent analyser and generator tools as well as the actual Hindi and English analysers and the Hindi generator. An overview of the major differences between Hindi and English is given in Section 5. This is followed by a detailed description of the syntactic and lexical-semantic divergences between Hindi and English from a computational linguistics perspective in section 6. Section 7 describes our experiences in developing an MT system using the UNL. Section 8 deals with issues of disambiguation in the system. The paper ends with conclusions and future directions in Section 8.

```
;======================== UNL ======================
;John who is the chairman of the company has arranged
;a meeting at his residence.
[S]
mod(chairman(icl>post):01.@present.@def,
    company(icl>institution):02.@def)
aoj(chairman(icl>post):01.@present.@def,
    John(icl>person):00)
agt(arrange(icl>do):03.@entry.@present.@complete.@pre
    d,John(icl>person):00)
pos(residence(icl>shelter):04,John(icl>person):00)
obj(arrange(icl>do):03.@entry.@present.@complete.@pre
    d,meeting(icl>conference):05.@indef)
plc(arrange(icl>do):03.@entry.@present.@complete.@pre
    d,residence(icl>shelter):04)
[/S]
;=====================================================
```

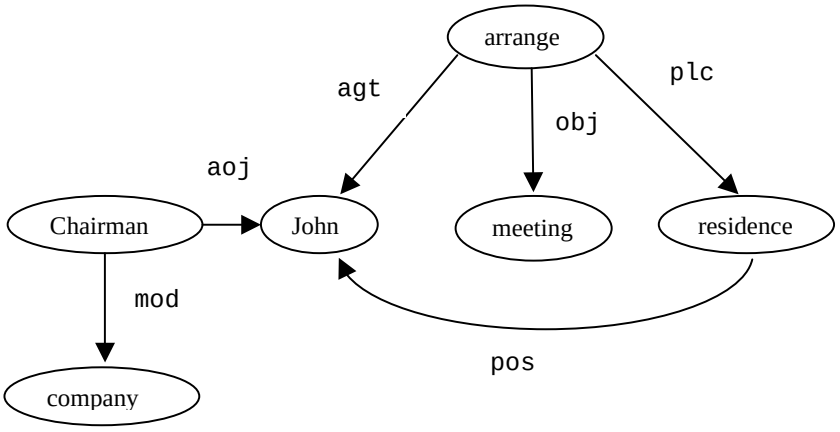


*Figure 1.* UNL expression and graph for example (3).

## 2. Conceptual Foundations

The strongest criticism against the interlingua based approach is that it requires the system designer to define a set of primitives which allow cross-language mappings. This task is looked upon as a very hard one (Vauquois and Boitet, 1985). Wilks says,

> *The notion of primitives in AI NL systems might be that they constitute not some special language, or another realm of objects, but are no more than a specialised sublanguage consisting of words of some larger standard language which plays a special organizing role in a language system.* (Wilks, 1987:759)

Since UNL is an interlingua we need to address this criticism. Rather than being based on primitives, the UNL system depends on a large repository of word concepts that occur in different languages. Such concepts are termed UWs. Thus words like *ikebana* and *kuchipudi* get included in this repository as `ikebana(icl>art form)` and `kuchipudi(icl>dance form)`. These word concepts are unambiguous, since every UW has a restriction that defines the sense of the basic UW used. For example, *spring* is a basic UW, which is disambiguated when it is restricted as `spring(icl>season)` meaning '*spring* included in the class of seasons'. The word concepts *spring* and *season* are ambiguous individually, but the combination `spring(icl>season)` is unambiguous. This can be further disambiguated as `spring(icl>(season(icl>time)))`.

No attempt is made in the UNL system to decompose concepts (acts, objects, states and manner) into primitives. A particular action, say *stab,* is represented using a single UW `stab(icl>do)`. This results in a representation that is more elegant and economical than some primitive based systems like Schank's CD.

## 2.1 THEORETICAL BACKGROUND

UNL expressions are made of binary relations. The RLs are designed to capture syntactic and semantic relations between UWs consistent with our knowledge of concepts and gathered from the corpus of languages. The relations are chosen keeping in mind the following principles:

**Principle 1. Necessary Condition**

The necessary condition is something that characterizes separate relations: a relation is necessary, if one cannot do without it.

**Principle 2. Sufficient Condition**

The sufficient condition characterizes the whole set of relations: the set meets this condition if one need not add anything to it.

**Explanation:**

Let $U=\{UW_1, UW_2, \ldots, UW_n\}$ be the UW lexicon
and $C=\{C_1, C_2, C_3, \ldots, C_m\}$ be the set of all possible contexts.
The set of RLs $\{RL_i\}$ in an interlingua *IL* defines functions of the following form:

$RL_i : U \times U \rightarrow C$

Let there be $p$ such RLs. We can call this set $R$ where,

$R=\{RL_1, RL_2, \ldots, RL_p\}$

Relating this to the UNL, $RL_1$ could be `agt`, $RL_2$ could be `obj`, $RL_3$ could be `ins` and so on. Also concretely, contexts could be subsets of all possible sentences in all languages at all times. Each $C_i$ is the set of all sentences in which each $RL_i$ consists of tuples of the form

$$\{((UW_{a_1},UW_{a_2}), C_a), ((UW_{b_1},UW_{b_2}), C_b)), \ldots\}$$

where every $((UW_{x_1},UW_{x_2}), C_x)$ is unique across the members of the set $R$. Each $C_x$ is the set of all possible sentences in which $UW_{x_1}$ and $UW_{x_2}$ appear. In this theoretical framework, contexts are language independent. Thus, the two equivalent sentences in (4) belong to the same context $C_q$, say.

(4) John is driving a car.
*John gaadi chalaa rahaa hai*
JOHN CAR DRIVE -ING IS

From this definition it is clear what the necessity and sufficiency conditions mean.

The necessity condition implies that if an RL $RL_x$ is removed from the inventory the corresponding set, $\{((UW_{a_1},UW_{a_2}), C_a), ((UW_{b_1},UW_{b_2}), C_b)), \ldots\}$ *cannot be expressed* in the IL.

Similarly the sufficiency condition implies that if we add another relation $RL_y$ then every element in the set $RL_y$ will be present in some existing set $RL_x$.

The UNL expressions are binary and do not include the context information that has been referred to in the above discussion. Actually, the UNL reflects the context information through the **semantic types** of the UWs and the RLs. For example, when we say `agt(`$UW_1$`,`$UW_2$`)`, it is clear that $UW_1$ is an event of which the volitional entity $UW_2$ is the agent. Thus, while encoding natural language sentences in the UNL, word and world knowledge will be used tor capture implicitly the context which has been described above in a hypothetical setting.

## 2.2 HOW UNIVERSAL IS THE UW LEXICON?

An obvious question that arises for the UWs is “Why call these universal, since they are based on English?”. However, Katz says:

> Although the semantic markers are given in the orthography of a natural language, they cannot be identified with the words or expressions of the language used to provide them with suggestive labels. (Katz, 1966:156)

This means that the primitives exist independently of the words used to describe, locate or interpret them. The Uws, though represented using Roman characters and English lexemes, are actually language-independent concepts.

However, a problem arises when a group of words has to be used in a language whose lexical equivalent is a single word in another language. For example, for the Hindi word देवर *devar* the English meaning is 'husband's younger brother'. Now, if we keep the universal word `husband's younger brother(icl>relative)` in the Hindi–UW dictionary and link it to *devar*, the analysis of the Hindi sentence shown in (5a) will produce a set of UNL expressions in which the UW `husband's younger brother(icl>relative)` appears. From this set, an English language generator generates the sentence (5b).

(5) a. laxman sita kaa devar hai
LAXMAN SITA-OF HUSBAND'S-YOUNGER-BROTHER IS
b. Laxman is Sita's husband's younger brother.

Now, the English analyser, while analysing (5b), will have the option of generating (6a) or (6b).

(6) a. `aoj(young(icl>state).@comparative, brother(icl>relative))`
`mod(brother(icl>relative), husband(icl>relative))`
b. `husband's younger brother(icl>relative)`

*Devar* was an example of conflation in noun for Hindi. For a verb, we can take *ausaanaa* which translates to English as 'to ripen by covering in straw'. Thus *ausaanaa* has a conflational meaning. The UW for this could be (7).

(7) [ausaanaa] `"ripen(met>cover(ins>straw))"`

Now if the UNL expressions contain the words *ripen*, *cover* and *straw* separately, then it is a non-trivial problem for the generator to produce the conflated verb *ausaanaa*. But if the above UW is used, then this can be done very easily.

One of the key assumptions about the UNL lexicon system is that the Language–UW (L-UW) dictionaries should be usable *without change* in both analysis and generation. However, as is apparent from the discussion above, achieving this kind of universality is an idealisation.

A general decision taken in the present work is to introduce the language-specific word as such in the UW dictionary, if the corresponding English description is long-winded and cumbersome. For example, we keep `kuchipudi(icl>dance)` in the dictionary instead of *an Indian dance form originating in the state of Andhra.* But, we do not keep `billi(icl>animal)`, where *billi* means 'cat' in Hindi, because `cat(icl>animal)` is available.

It should be noted that, the headwords are not always English words. Roman letters are used to represent all the concepts that are found in all the languages at all times. Thus, *ikebana* and *kuchipudi* which are not English words are also stored in the dictionary. The disambiguation is done by a construct called the **restriction**. Restrictions are written in Roman letters. But they do not depend on English. The senses are not the ones that are peculiar to the English language. For example, one of the senses found in India of the word *back bencher* is 'student who is not serious in his/her studies and whiles away the time sitting at the back of the class'. This additional sense is included in the UW dictionary as `back-bencher(icl>student)`. Thus if a particular word *w* in English has acquired an additional sense in another language, this sense is introduced into the UW dictionary by tagging the appropriate restriction. The words in specific languages get mapped to specific word senses and not to the basic UWs. The basic UWs are ambiguous and the linking process is carried out only after disambiguating.

We have given the example of *devar* 'husband's younger brother' in Hindi. This illustrates the case where there is no direct mapping from Hindi to an English word. We have to discuss the reverse case where for an English word there is no direct mapping in another language. This is important since the UWs are primarily constructed from English lexemes. We have decided that if an English word is commonly used in Hindi, we keep the Hindi transliterated word in the dictionary. For example, for the word *mouse* used in the sense of an input device for the computer we keep (8) in the lexicon.

(8) [maausa] `"mouse(icl>device)"`

The same strategy is adopted if a word is very specific to a language and culture. For example, for the English word *blunderbuss* (an old type of gun with a wide mouth that could fire many small bullets at short range),

there is no simple Hindi equivalent and so we keep the transliteration (9) in the lexicon.

(9) [blaandarbasa] `"blunderbuss(icl>gun)";`

The topic of multiple words for 'snow' in Eskimo languages is very popular in the NLP, MT and Lexical Semantics literature. We have discussed how to link these words with the appropriately formed UWs. In the Eskimo language Inuit, the following are a few examples for the word 'snow': *aput* 'snow (in general)', *pukak* 'snow (like salt)', *mauja* 'soft deep snow', *massak* 'soft snow', *mangokpok* 'watery snow'.

The rich set of RLs of UNL are exploited to form the UWs which in this case respectively are shown in (10).

(10)
```
[aput] "snow(icl>thing)";
[pukak] "snow(aoj<salt like)";
[mauja] "snow(aoj<soft, aoj<deep)";
[massak] "snow(aoj<soft)";
[mangokpok] "snow(aoj<watery)";
```

Note the disambiguating constructs for expressing the UWs. The RLs of the UNL are used liberally. `aoj` is the label for the adjective–noun relation.

The issue of shades of meaning is a very important one, and the main idea again is that the RLs of UNL can be used in the lexicon too. In (11) we show are some examples of the verb *get off* and in (12) the noun *shadow*. (The gloss sentences are attached for clarifying the meaning, which anyway gets communicated through the restrictions)

(11) [prasthaana karanaa] `"get off(icl>leave)";` We got off after breakfast

[bacanaa] `"get off(icl>be saved)";` lucky to get off with a scar only

[bhejanaa] `"get off(icl>send)";` Get these parcels off by the first post

[bandha karanaa] `"get off(icl>stop)";` get off the subject of alcoholism

[kaama rokanaa] `"get off(icl>stop,obj>work)";` get off (work) early tomorrow.

(12) [andhera] `"shadow(icl>darkness)";` the place was now in shadow

[kaalii dhabbaa] `"shadow(icl>patch)";` shadows under the eyes.

[paraCaai[] `"shadow(icl>atmosphere)";` country in the shadow of war

[saMkot] "shadow(icl>hint)" ; the shadow of things to come

[saayaa] "shadow(icl>close company)"; the child was a shadow of her mother

[Caayaa] "shadow(icl>deterrant)"; a shadow over his happiness

[SaraNa] "shadow(icl>refuge)"; he felt secure in the shadow of his father

[aabhaasa] "shadow(icl>semblance)"; shadow of power

[bhuuta] "shadow(icl>ghost)"; seeing shadows at night

Again, note should be made of how the restrictions disambiguate and address the meaning shade.

## 2.3 POSSIBILITY OF REPRESENTATIONAL VARIATIONS

Another important consideration while accepting UNL as an interlingua is the way it represents a particular sentence. UNL gives an unambiguous semantic representation of a sentence, but it does not claim uniqueness of the representation. Justifying the need for primitives in an Interlingua, Hardt (1987:196) says, “The requirement that sentences that have the same meaning be represented in the same way cannot be satisfied without some set of primitive ACTs”. This requirement may be a necessary condition for a knowledge-representation scheme, but surely not for an interlingua. For example, consider the sentences in (13).

(13) a. John gave a book to Mary.
b. The book was given by John to Mary.
c. Mary received a book from John.
d. Mary took a book from John.

All these sentences have similar meanings, but are different from the point of view of the stylistics, focus and aspect. This is reflected in the corresponding UNL representations shown in (14). As shown in (14b), `@topic` is used for sentences in passive form to give more importance to the object than to the subject.

(14) a.
```
[S] agt(give(icl>do).@entry.@past,John(icl>person))
obj(give(icl>do).@entry.@past,book(icl>text).@def)
ben(give(icl>do).@entry.@past,Mary(icl>person))
[/S]
```

b.
```
[S] agt(give(icl>do).@entry.@past,
   John(icl>person))
obj(give(icl>do).@entry.@past,
   book(icl>text).@def.@topic)
ben(give(icl>do).@entry.@past,Mary(icl>person))
[/S]
```

c.
```
[S] agt(receive(icl>do).@entry.@past,
      Mary(icl>person))
obj(receive(icl>do).@entry.@past,
   book(icl>text).@def)
src(receive(icl>do).@entry.@past,John(icl>person))
[/S]
```

d.
```
[S] agt(take(icl>do).@entry.@past,
      Mary(icl>person))
obj(take (icl>do).@entry.@past,
      book(icl>text).@def)
src(take(icl>do).@entry.@past, John(icl>person))
[/S]
```

Using these UNLs, a generator can generate an exact translation of the respective sentences and not its paraphrase, as happens with CD-based generators.

Although UNL *represents* similar information in different ways as above, its utility as a knowledge-representation scheme does not get affected. Seniappan and Bhattacharyya (2000) have investigated the use of UNL for automatic intra-document hypertext linking and have claimed that their system has an ability to extract anchors which are relevant but do not surface when frequency based methods are used.

As a summary of this section on conceptual foundations we mention the following points:

1. The UNL system strives to achieve language independence through its vast and rich repository of universal words.
2. The basic UWs, i.e. the unrestricted headwords, are mostly English words. But this does not make the UW dictionary an English language lexicon, since the concepts denoted by these UWs are valid for all languages.
3. Whenever a language-specific word is cumbersome to express in English, the word is introduced into the UW repository after placing the proper restriction that clarifies the meaning of the particular UW and classifies it in a particular domain.
4. The RLs have stabilised to 41 and seem adequate to capture semantic relations between concepts across all languages. This is,

however, only an empirical statement keeping in mind the necessity and the sufficiency conditions.

5. A large portion of the burden of expressiveness in the UNL is carried by the attribute labels that indicate how the word is used in the sentence.
6. The UW repository is the union of all concepts existing in all languages at all times.

## 3. L-UW Dictionary and the Universal Lexicon

In this section, we discuss the structure of an L-UW dictionary, its language-dependent and -independent parts and the associated attributes. The restriction attached to every word not only disambiguates it, but also puts it under a predefined hierarchy of concepts, called the "knowledge base" in the UNL parlance. To construct the L-UW dictionary, the UWs are linked with the language words. Morphological, syntactic and semantic attributes are then added. For example, for the UW `dog(icl>mammal)`, the Hindi word ku%ta *kutta* 'dog' is the language word, the morphological attribute is `NA` (indicating word ending with Aa), the syntactic attribute is `NOUN` and the semantic attribute is `ANIMATE`. A part of the entry is (15).

(15) [ku%ta] `"dog(icl>mammal)" (NOUN, NA, ANIMATE);`

The language-independent part of this entry are `dog(icl>mammal)` and `ANIMATE`, while the language-dependent parts are ku%ta and `NA`. The same L-UW dictionary is used for the analysis and the generation of sentences for a particular language.

### 3.1 ARCHITECTURE OF THE L-UW DEVELOPMENT SYSTEM

Figure 2 shows the architecture of the L-UW development system with both language-dependent and language-independent components. The language-independent parts are the ontology space and the set of Uws. The language-dependent parts are the language-specific dictionary and the syntactic and morphological attributes.

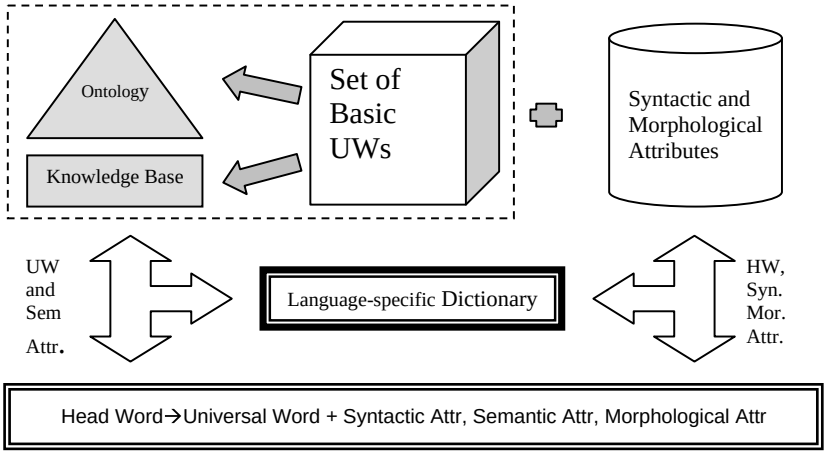


*Figure 2.* Integrated system for Language-UW Lexicon building

The process of L-UW dictionary construction can be partially automated. This achieves accuracy and exhaustiveness. Lexicon developers find it difficult manually, consistently and exhaustively to insert the hundreds of semantic attributes required for the accurate analysis of the sentences. Also it is difficult to achieve uniformity in putting the restrictions. For example, for the noun *book*, a lexicon developer may restrict the meaning of *book* as `book(icl>concrete thing)`, `book(icl>textbook)`, `book(icl>register)`, etc. This leads to non-uniformity in the UWs which can be avoided by standardizing the knowledge base, i.e. the UW repository. A brief description of the various components of the dictionary construction system now follows.

### 3.1.1 *Language-independent Components*

**The Ontology Space**

The Ontology Space refers to a hierarchical classification of the word concepts. This ontology is in the form of a Directed Acyclic Graph (DAG). Our system uses the upper CYC Ontology (Guha et al., 1990) which has around 3,000 concepts. This ontology is language independent and provides the semantic attributes.

**The Set of UWs or the Knowledge base**

The set of basic UWs, i.e. the unrestricted Uws, contains mostly the root words of the English language. Also, there are words from other languages, which do not have simple English equivalents, e.g. *ikebana* from Japanese and *kuchipudi* from Telugu. Basic UWs generally have more than one

meaning. They are disambiguated by adding restrictions. These restricted UWs are language independent. A new knowledge base is in the process of being introduced and the UWs will be drawn from this resource.

### 3.1.2 *Language-dependent Components*

**Language-specific Word Dictionary**

After selecting the UW, the corresponding language-specific string is found by consulting the dictionary of the particular language and by translating the gloss attached.

**Syntactic and Morphological Attributes**

This set includes attributes like part of speech, tense, number, person, gender, etc. and morphological attributes which describe paradigms of morphological transformations. These attributes are language specific and are inserted by the lexicon developer.

## 3.2 CONSTRUCTING DICTIONARY ENTRIES

The procedure of constructing dictionary entries is partially automated as follows:

1. The human expert selects a UW from the knowledge base and finds for this sense the position of the basic UW (the portion left after stripping the restriction) as a leaf in the ontology. Consider a snapshot of the CYC ontology DAG given in Figure 3. Suppose we want to make a dictionary entry for the word *animal.* The word *is* found as a leaf in the ontology. The UW is `animal(icl>living thing)`.
2. The semantic attributes of this UW are the nodes traversed while following all paths from the leaf to the root (`thing` in this case). For example, the following attributes are generated for the word *animal* = `SolidTangibleThing, TangibleThing, PartiallyTangible, PartiallyIntangible, CompositeTangibleAndIntangibleObject, AnimalBLO, BiologicalLivingObject, PerceptualAgent, IndividualAgent, Agent, Organism-Whole, OrganicStuff, SomethingExisting, TemporalThing, SpatialThing, Individual, Thing`

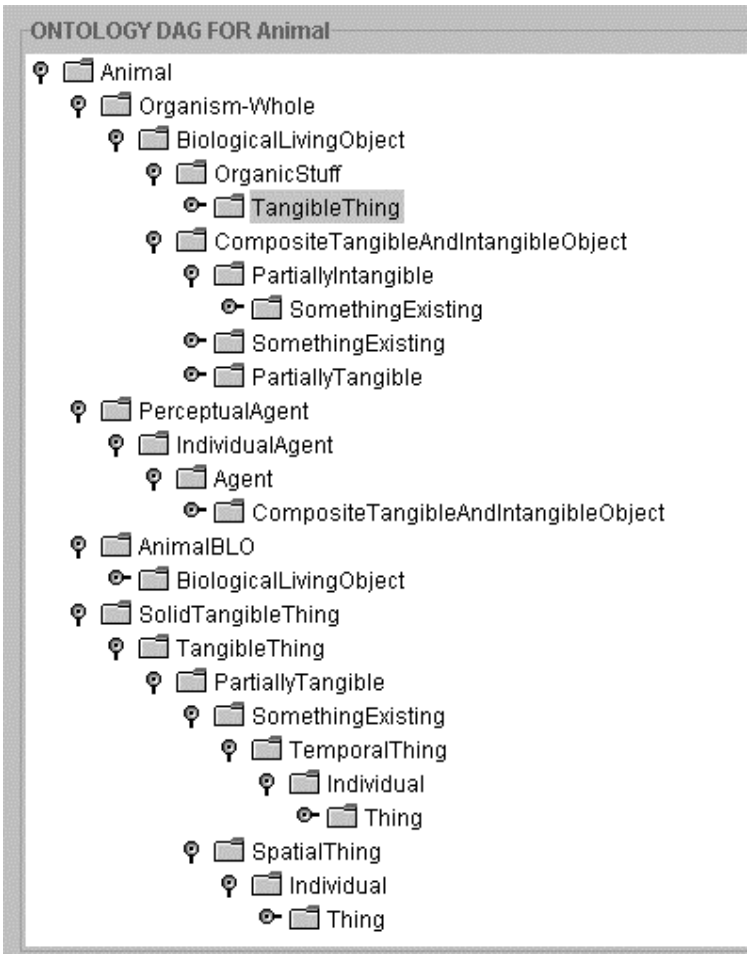


*Figure 3.* A Snapshot of the CYC Upper-level Ontology

3. The work of the human expert is now limited to adding the syntactic and morphological attributes. These attributes are far less in number than semantic attributes. Thus, the labour of making semantically rich dictionary entries is reduced.

An example of a dictionary entry generated by the above process is shown in (16).

(16) [praaNii] `{ }"animal(icl>organism whole)"(Noun, NI, SolidTangibleThing, TangibleThing, PartiallyTangible, PartiallyIntangible, CompositeTangibleAndIntangibleObject, AnimalBLO, BiologicalLivingObject, PerceptualAgent, IndividualAgent, Agent, Organism-Whole, OrganicStuff, SomethingExisting, TemporalThing, SpatialThing, Individual, Thing)`

*praanee* is the Hindi equivalent for *animal.* `Noun` and `NI`[1] are the syntactic and morphological attributes added by the human lexicon developer.

## 4. The System

We describe here the systems we built, viz. the Hindi analyser which converts Hindi sentences into UNL expressions, the English analyser which produces UNL expressions from English sentences and the Hindi generator which generates Hindi sentences from UNL expressions. The analysers use a software called the EnConverter while the generator uses the DeConverter.[2] These tools are language-independent systems that are driven by the language-dependent rule base and the L-UW dictionaries. We first give an overview of the working of the EnConverter and DeConverter engines. Then we explain in brief the three systems. Space restriction does not permit detailed description of all three systems.

### 4.1 THE ANALYSER MACHINE

The EnConverter is a language-independent analyser that provides a framework for morphological, syntactic and semantic analysis synchronously. It analyses sentences by accessing a knowledge-rich **L-UW lexicon** and interpreting the **analysis rules**. The process of formulating the rules is in fact programming a sophisticated symbol-processing machine.

The EnConverter can be likened to a multi-head Turing machine. Being a Turing Machine, it is equipped to handle phrase-structure (type 0) grammars (Martin, 1991) and consequently the natural languages. The EnConverter delineates a sentence into a tree, called the "nodenet tree", whose traversal produces the UNL expressions for the sentence. During the analysis, whenever a UNL relation is produced between two nodes, one of these nodes is deleted from the tape and is added as a child of the other

node to the tree. It is important to remember this basic fact to be able to understand the UNL generation process in myriad situations.

The EnConverter engine has two kinds of heads: **processing heads** and **context heads**. There are two processing heads, called "analysis windows". The nodes under these windows are processed for linking by a UNL RL and/or for attaching UNL attributes to. A node consists of the language-specific word, the UW and the attributes appearing in the dictionary as well as in the UNL expressions. The context heads are located on either side of the processing heads and are used for look ahead and look back. The machine has functions like shifting the windows right or left by one node, adding a node to the node-list (tape of the Turing machine), deleting a node, exchange of nodes under processing heads, copying a node and changing the attributes of the nodes. The complete description of the structure and working of the EnConverter can be found in UNU (2000b).

## 4.2 THE ENGLISH ANALYSER

The English analyser makes use of the English–UW dictionary and the rule base for English analysis, which contains rules for morphological, syntactic and semantic processing. At every step of the analysis, the rule base drives the EnConverter to perform tasks like completing the morphological analysis (e.g. combine *boy* and *'s*), combining two morphemes (e.g. *is* and *working*) and generating a UNL expression (e.g. `agt` relation between *he* and *is working)*. Many rules are formed using context-free grammar-like segments, the productions of which help in clause delimitation, prepositional-phrase (PP) attachment, part-of-speech disambiguation and so on. This is illustrated with an example of noun clause handling (17), which is handled by the grammar in (18).

(17) The boy who works here went to school.

(18)
```
CL→ V            ; e.g. The boy who works …
   | ADV V N     ; e.g. … who fluently speaks English
   | V ADV       ; e.g. … who works here
   | V ADV ADV   ; e.g. … who ran very quickly
```

The processing goes as follows.

1. The clause *who works here* starts with a relative pronoun and its end is decided by the system using the grammar. There is no rule like CL( V ADV V and so the system does not include went in the subordinate clause.

2. The system detects here as an adverb of place from the lexical attributes and generates plc (place relation) with the main verb work of the subordinate clause. After that, work is related to boy through the agt relation. At this point the analysis of the clause finishes.
3. boy is now linked with the main verb went of the main clause. Here too the agt relation is generated.
4. The main verb is then related with the preposition phrase to generate plt (indicating "place to"), taking into consideration the preposition to and the noun school (which has PLACE as a semantic attribute in the lexicon). The analysis process thus ends.

A typical example of the ability of the system to disambiguate parts of speech is shown in the UNL representation for (19) in Figure 4.

(19) The soldier went away to the totally deserted desert to desert the house in the desert.

```
======================== UNL ======================
The soldier went away to the totally deserted desert
to desert the house in the desert
[S]
mod(deserted(icl>vacant):11,total(icl>complete):0T)
aoj(deserted(icl>vacant):11,desert(icl>landscape):1A.@
  def)
plc(go(icl>event):0C.@entry.@past.@pred,
  away(icl>logical place):0H)
obj(desert(icl>do):1K.@present.@pred,house(icl>place):
  1V.@def)
plc(desert(icl>do):1K.@present.@pred,
  desert(icl>landscape):28.@def)
plt(go(icl>event):0C.@entry.@past.@pred,
  desert(icl>landscape):1A.@def)
pur(go(icl>event):0C.@entry.@past.@pred,
  desert(icl>do):1K.@present.@pred)
agt(go(icl>event):0C.@entry.@past.@pred,
  soldier(icl>human):04.@def)
[/S]
;===================================================
```

*Figure 4.* Example of part-of-speech disambiguation

The adjectival form of *desert* is represented as `deserted(icl>vacant)`. The noun form is `desert(icl>landscape)`, while the verb form is `desert(icl>do)`. The analysis rules make use of

the linguistic clues present in the sentence. Thus, the adverb *totally* preceded by the article *the* makes *deserted* an adjective, which in turn makes the following *desert* a noun.

The system can also convert sentences in which relative pronouns do not occur in the sentence explicitly, for example (20).

(20) a. The study (which was) published in May issue was exhaustive.
b. He lives at a place (where) I would love to be at.
c. He gave me everything (that) I asked for.
d. The cabbage (which was) fresh from the garden was tasty.

Various heuristics are used to decide the start of clause and the relative pronoun that is implicit. Some of these are:

- Presence of two verbs with a single subject as in (209a).
- A noun followed by a pronoun as in (20b).
- Quantifiers like *all, everything* and *everyone* followed by another pronoun or noun as in (20c).
- An adjective following a noun as in (20d).

Semantic attributes stored in the dictionary are exploited to solve ambiguities of PP and clausal attachment as exemplified in (21).

(21) a. He went to my home when I was away.
b. He met me at a *time when I was very busy*.

The structures of the two sentences are similar, but semantic attributes indicate that *when* qualifies temporal nouns like *time, hour, second*, etc. Thus, in (21a) the system attaches the clause *when I was away* to the verb considering it an adverb clause of time, while in (21b) it attaches the clause *when I was very busy* to the noun considering it an adjective clause.

Anaphora resolution is dealt with in a limited way at the sentence level. This can be seen from the UNL expressions produced by the system for (22) as shown in Figure 5.

(22) He built his house in a very short span of time.

The UW-IDs (a form of identifier) of both the instances of `he(icl>person)` in (22) are the same, viz. `:09`. The system does not do the same for (23), since it is not certain whether *John* and *he* refer to the same person.

(23) John built his house.

Ellipsis handling is done for various kinds of sentences. A few examples are in (24).

(24) a. I reached there before he could (reach).
b. (I am) Sorry, I did it.
c. I went to Bombay and then (I went) to Delhi.

```
;======================== UNL ======================
;He built his house in a very short span of time.
[S]
mod(house(icl>place):0D, he(icl>person):09)
agt(build(icl>event):03.@entry.@past.@pred,
he(icl>person):09)
mod(short(icl>less):0T,very:0O)
aoj(short(icl>less):0T,span(icl>duration):0Z.@indef)
obj(built(icl>event):03.@entry.@past.@pred,
house(icl>place):0D)
dur(built(icl>event):03.@entry.@past.@pred,
span(icl>duration):0Z.@indef)
mod(span(icl>duration):0Z.@indef,time(icl>abstract
thing):AB)
[/S]
;===================================================
```

*Figure 5.* UNL representation for sentence (22).

For (24a), the implicit *reach* is produced explicitly in the UNL expressions. (24b) obviously does not generate an extra *I*, but adds the attribute `@apology` to the verb *do*. Since there are two events of `going` in (24c), an explicit *go* is produced but not the extra *I* as the agent is the same for both the instances of *go*.

Thus, the English analyser is capable of handling many complex phenomena of the English language. The system also can guess a UW for a word not present in the lexicon. Currently, it has around 5,800 rules. A detailed explanation of the system can be found in Parikh et al. (2000) and Parikh and Bhattacharyya (2001).

## 4.3 THE HINDI ANALYSER

The rule base that drives the Hindi analyser uses strategies different from its English counterpart. This is due to the numerous structural differences between Hindi and English (see Section 5). But the fundamental mechanism of the system is the same, i.e. it performs morphological, syntactic and semantic analysis synchronously.

The rule base of the Hindi analyser can be broadly divided into three categories: morphological rules, composition rules and relation resolving

rules. Morphology rules have the highest priority. This is because unless we have the morphed word, we cannot decide upon the part of speech of the word and its relation with the adjacent words. Hindi has a rich morphological structure. Information regarding person, number, tense and gender can be extracted from the morphology of nouns, adjectives and verbs. An exhaustive study of the morphology is done for this purpose and appropriate rules are incorporated into the system (Monju et al., 2000). To illustrate the process of Hindi analysis, consider the Hindi sentence (25) which has an explicit pronoun.

(25) maine dekhaa ki seetaa sabjee khareed rahee hai
I SAW THAT SITA VEGETABLE BUY -ING IS
'I saw that Sita is buying vegetables.'

The processing of this sentence is carried out as follows:

1. The beginning of the clause is marked by the presence of the relative pronoun *ki* 'that'.
2. The analysis windows right shift until the predicate *dekhaa* 'saw' is reached.
3. All the relations of the previous nodes with this predicate are resolved. In this case, *mai* 'I' being a first person singular and animate pronoun, `agt` relation is produced between *maine* and *dekhaa.*
4. The relative pronoun *ki* is now detected and the analysis heads right shift. It combines *ki* with *dekhaa* and adds a dynamic attribute `kiADD` to *dekhaa*.
5. The clause following *ki* is now resolved. The analysis windows right shift until the main predicate of the sentence, *khareed rahee hai* 'is buying' is reached.
6. It combines the nodes *sabjee* 'vegetables' and *khareed rahee hai* with the `obj` relation seeing the inanimate attribute of *sabjee.*
7. It then resolves the `agt` relation between *seetaa* 'Sita' and *khareed rahee hai* seeing the animate attribute of *seetaa*.
8. At the end of its analysis, its main predicate is retained which in this case is *khareed rahee hai*. Finally the `obj` relation is generated between this verb and *dekhaa*.

Composition rules are used to combine a noun or a pronoun in a sentence with a postposition or case-marker following it. During combination, the case marker is deleted from the node list and appropriate attributes are added to the noun or pronoun to retain the information that the particular noun or pronoun had a postposition marker following it. For example, consider the sentences pairs (26)–(29).

(26) raam ne raavan ko teer se maaraa
RAAM-subj RAAVAN-TO ARROW-WITH KILLED
'Raam killed Raavan with an arrow.'

(27) ped se patte baag mein geere
TREE-FROM LEAVES GARDEN-IN FELL
'Leaves fell in the garden from the trees.'

(28) peeTar subah se kaam kar rahaa hai
PETER MORNING-SINCE WORK DO -ING IS
'Peter has been working since the morning.'

(29) bachche se taalaa khulaa
CHILD-BY LOCK WAS-OPENED
'The lock was opened by the child.'

In (26)–(29), *teer* 'arrow', *ped* 'tree', *subah* 'morning' and *bachchaa* 'child' are nouns and are followed by the same postposition marker से *se* 'with/from/since/by'. However, as is evident from the English translation, the meaning of *se* is different in each sentence. Hence, the noun preceding it forms a different relation with the main verb in each case as in (30).

(30) a. `ins(kill(icl>do).@past, arrow(icl>thing))`
b. `plf(fall(icl>occur).@past, tree(icl>place))`
c. `tmf(work(icl>do).@present,@progress, morning(icl>time))`
d. `agt(open(icl>do).@past, child(icl>person))`

These nouns have the semantic attributes `INSTRU` (can be used as an instrument), `PLACE`, `TIME` and `ANI` (animate entities) respectively in the lexicon. They help to decide the sense of the case marker and thus the role of the noun in the particular sentence. When the case marker *se* is combined with the noun preceding it, attributes `INS` (instrument), `PLF` (place from which an event occurs), `TMF` (time from which an event has started) and `AGT` (agent of the event), are added to the respective nouns. These attributes then lead to the production of the UNL relations shown in (30) for sentences (26)–(29) respectively.

Now we describe the various Hindi-language phenomena handled by the system. Hindi is a null-subject language (see Section 6.1.4]. This means that it allows the syntactic subject to be absent. For example, sentence (31) is valid in Hindi.

(31) jaa rahaa hun
GOING AM
* ‘am going’

The system makes the implicit subject explicit in the UNL expressions. The procedure to do this is discussed in Section 6.1.4. The UNL expression produced by the system in this case is (32).

(32)
```
[S]
agt(go(icl>do).@entry.@present.@progress,
    I(icl>person))
[/S]
```

The system can also handle limited amount of anaphora resolution. For example, consider the sentence in (33a) and the corresponding UNL relations generated as shown in (33b).

(33) a. meree ne apanee kitaab jeem ko dee hai
MARY-subj HER BOOK JIM-TO HAS-GIVEN
‘Mary has given her book to Jim.’

b.
```
[S]
pos(book(icl>publication):0C,Mary(icl>person):00)
ben(give(icl>do):0R.@entry.@present.@pred,
   Jim(icl>person):0J)
obj(give(icl>do):0R.@entry.@present.@pred,
   book(icl>publication):C)
agt(give(icl>do):0R.@entry.@present.@pred,
   Mary(icl>person):00)
[/S]
```

That resolution of the anaphora is apparent from the fact that the UW `she(icl>person)` for *her* is replaced by `Mary(icl>person)` in the `pos` relation.

One of the major differences between Hindi and English is that a single pronoun *vah* in Hindi is mapped to two pronouns *he* and *she* of English. The gender of the pronoun in Hindi can be known only from the verb morphology. So the system defers the generation of the UW for *vah* until the verb morphology is resolved. At the end of the analysis, the correct `he(icl>person)` or `she(icl>person)` is produced, for example (34).

(34) a. vah shaam ko aaegee
HE/SHE EVENING-IN WILL-COME(fem)
‘She will come in the evening.’

b.
```
[S]
tim(come(icl>do):0D.@entry.@future,
  evening(icl>time):05.@def)
```

```
agt(come(icl>do):0D.@entry.@future,she(icl>person):
  00)
[/S]
```

Hindi uses the word-forms Aaegaa *aaegaa* and AaegaI *aaegee* for the future of the verb Aa *aa* 'come' for a male subject and female subject respectively. Thus, in (34a), the verb *aaegee* causes the UW `she(icl>person)` to be generated for *vah*.

Hindi being a relatively free word-ordered language, the same sentence can be written in more than one way by changing the order of words, as in (35a–c) for example. The output in all three cases is (35d).

(35) a. tum kahaan jaa rahe ho?
YOU WHERE GO -ING ARE

b. kahaan tum jaa rahe ho?
WHERE YOU GO -ING ARE

c. kahaan jaa rahe ho tum?
WHERE GO -ING ARE YOU

'Where are you going?'

d.
```
[S]
plc(go(icl>do):07.@entry.@interrogative.@pred.
  @present.@progress, where(icl>place):00)
agt(go(icl>do):07.@entry.@interrogative.@pred.
  @present.@progress, you(icl>male):0I)
[/S]
```

This is achieved as follows. Additional rules are added for each combination of the word types. Also the rules are prioritised such that the right rules are picked up for specific situations. For the sentence (35a), first the rule for generating a `plc` relation between *kahaan* and *jaa rahe ho* is fired, followed by the rule for generating the `agt` relation between *tum* and *jaa rahe ho*. In (35b), first `agt` and then `plc` are resolved. In (35c), a rule first exchanges the positions of *jaa rahe ho* and *tum*. After that the rules fire as before for setting up the relations. Use is made of the question mark at the end of the sentence.

Hindi allows two types of constructions for adjective clauses: one with explicit clause markers like jaao *jo* 'who', ijasakI *jisakee* 'whose', ijasao *jise* 'whom', etc. and the other with the vaalaa *vaalaa* 'ing' construction. Our analyser can handle both (36a,b). The system produces the same UNL relations (36c) for both these.

(36) a. peeTar jo london mein rahataa hai vah yahaan kaam karataa hai

PETER WHO LONDON-IN STAYS HE HERE WORK-DO-IS

'Peter who stays in London works here.'

b. london mein rahanevaalaa peeTar yahaan kaam karataa hai

LONDON-IN STAYING PETER HERE WORK-DO-IS

Peter who stays in London works here.

c.
```
[S]
agt(work(icl>do).@entry.@present,
   Peter(icl>person))
plc(work(icl>do) .@entry.@present, here)
agt(stay(icl>do) .@present, Peter(icl>person))
plc(stay(icl>do) .@present, London(icl>place))
[/S]
```

The two incoming arrows into `Peter(icl>person)` provide the clue to the system to identify correctly the adjective clause in each sentence.

Unlike English, Hindi has a way of showing respect to a person (see Section 5). This is conveyed through the verb morphology (37).

(37) mere chaachaa padh rahe hai

MY UNCLE READ -ING ARE

'My uncle is reading.'

The verb form here is for the subject in plural form. But since *uncle* is singular, the system infers that the speaker is showing respect and generates the `@respect` attribute for `uncle(icl>person)`.

The Hindi analyser can deal with simple, complex, compound, interrogative as well as imperative sentences. Currently the number of rules in the Hindi analyser is about 3,500 and the lexicon size is around 70,000.

### 4.4 THE GENERATOR MACHINE

The DeConverter is a language-independent generator that provides a framework for morphology generation and syntax planning synchronously. It generates sentences by accessing a knowledge-rich L-UW dictionary and interpreting the generation rules.

The working and the structure of the DeConverter are very similar to that of the EnConverter. It processes the UNL expressions on the input tape. It traverses the input UNL graph and generates the corresponding target-language sentence. Thus, during the course of the generation, whenever a UNL relation is resolved between two nodes, one of the nodes is inserted into the tape.

Like the EnConverter, the DeConverter also has two types of heads: **processing heads** and **context heads**. There are two processing heads, called “generation windows”, and only the nodes under these take part in any generation tasks like the left or right placement of the words and the resolution of attributes into morphological strings. The context heads, called the “condition windows”, are located on either side of the processing heads and are used for look ahead and look back. The machine has functions of shifting right or left by one node, adding a node to the node-list (tape of the Turing machine), deleting a node, exchange of nodes under processing heads, copying a node and changing attributes of the nodes. The complete description of the structure and working of the DeConverter can be found in UNU (2000a).

## 4.5 HINDI GENERATOR

The Hindi generator attempts to generate the most natural Hindi sentence from a given set of UNL expressions. The generation process is based on the predicate-centric nature of the UNL. It starts from the UW of the main predicate and the entire UNL graph is traversed in stages producing the complete sentence. The rule base contains the syntax planning rules and the morphology rules. Syntax planning is in general achieved with a very high degree of accuracy using two fundamental concepts called “parent–child relationships” and “matrix-based priority of relations” (D’Souza et al., 2001).

In a UNL relation `rel(`$UW_1$`,`$UW_2$`)`, the $UW_1$ is always the parent node and $UW_2$ the child. The syntax-planning task is to decide upon the right or left insertion of the child with respect to its parent. The UNL specification puts constraints on the possible types of UWs that can occur as $UW_1$ and $UW_2$ of a particular relation. Using this information and the relation between the two UWs, the position of the child relative to the parent is arrived at.

Another important consideration is the traversal of the UNL graph. The path is decided based on the relative priority of UNL relations which is in turn decided by the priority matrix. An example matrix is given in Table I. Such an exhaustive matrix is produced for all the 41 relations.

*Table I.* An example priority matrix, where L means `placed-left-of` and R means `placed-right-of`.

| | agt | obj | ins |
|---|---|---|---|

| | | | |
|---|---|---|---|
| agt | - | L | L |
| obj | R | - | R |
| ins | R | L | - |

According to the matrix in Table I, child(agt) is the leftmost element, child(ins) is the middle element and child(obj) is the rightmost element of the three. For example, consider the UNL expressions in (38a). The sentence generated according to Table I is (38b).

(38) a.
```
[S]
agt(eat(icl>do).@entry.@past, Mary(icl>person))
ins(eat(icl>do).@entry.@past,
    spoon(icl>thing).@indef)
obj(eat(icl>do).@entry.@past, rice(icl>food))
[/S]
```
b. meree ne chammach se chaaval khaayaa
MARY-subj SPOON-WITH RICE ATE
'Mary ate the rice with a spoon.'

The rule writer uses the matrix in Table I to decide upon the priorities of the rules. The relation for which the child is placed leftmost in the sentence has the highest priority and is resolved first, while the relation for which the child is placed rightmost, i.e. nearest to the verb, has the lowest priority.

Morphology generation not only transforms the target-language words for each UW, but also introduces case markers, conjunctions and other morphemes according to the RLs, a procedure reified as relation label morphology. Table II gives an idea of this process. UNL attributes reflecting the aspect, tense, number, etc. also play a major role in the morphology processing.

*Table II.* RL Morphology. "Position" indicates position of the word w.r.t. child (*M*)

| Relation *M* | Position | Word to be introduced |
|---|---|---|
| Agt | L | *ne* |
| And | R | *aur* 'and' |
| Bas | L | *se* 'as compared to' |
| Cag | L | *ke saath* 'with' |
| Cob | L | *ke saath* 'with' |
| Con | L | *yadi* $UW_2$ *to* $UW_1$ (if $UW_2$ then $UW_1$) |
| Coo | R | *aur* 'and'/ null |
| Fmt | R | *se* 'to' |

| | | |
|---|---|---|
| `Gol` | L | *mein* ‘into’ |
| `Ins` | L | *se* ‘using’ |
| `Mod` | L | *kaa* / *ke* / *kee* ‘of’/ null (depends on gender and number) |

The Hindi generator can produce both complex and compound sentences. The presence of a clause in the sentence is detected in two different ways: (i) presence of a “scope”, i.e. a compound UW which is a label for more than one UNL expression or (ii) presence of two incoming arrows from two different predicates. For example, (39a), can be represented in the UNL in two different ways (39b,c). In (39b), `boy(icl>person)` has two incoming arrows from `scold(icl>do)` and `hit(icl>do)`. (39c) explicitly marks the presence of the clause using the scope `:01`. The system generates the same sentence for both representations.

(39) a. He scolded the boy who had hit John.

b.
```
[S]
agt(scold(icl>do).@past.@entry, he(icl>person))
obj(scold(icl>do.@past.@entry, boy(icl>person))
agt(hit(icl>do).@pred.@complete.@past,
  boy(icl>person))
obj(hit(icl>do).@pred.@complete.@past,
  John(icl>person))
[/S]
```

c.
```
[S]
agt(scold(icl>do).@past.@entry, he(icl>person))
obj(scold(icl>do).@past.@entry, :01)
agt:01(hit(icl>do).@pred.@complete.@past.@entry,
  boy(icl>person))
obj:01(hit(icl>do).@pred.@complete.@past.@entry,
  John(icl>person))
[/S]
```

The Hindi generator is also capable of handling imperative, passive and interrogative sentences. The current system has around 5,000 rules and uses the same Hindi–UW dictionary used by the Hindi analyser.

## 5. Major Differences between Hindi and English

The basic difference between Hindi and English is the sentence structure. Hindi has an SOV structure for sentences, while English follows the SVO

order. Rao et al. (2000) give the structure shown in (40a) for English sentences where, S is subject, O object, V verb, $S_m$ subject post-modifiers, $O_m$ object post-modifiers, $V_m$ the expected verb post-modifiers and $C_m$ the optional verb post-modifiers, exemplified in (40b).

(40) a. S $S_m$ V $V_m$ O $O_m$ $C_m$

b. [The President]$_S$ [of America]$_{Sm}$ [will visit]$_V$ [the capital]$_O$ [of Rajasthan]$_{Om}$ [in the month of December]$_{Cm}$.

On the other hand, Hindi has the structure shown in (41a), as illustrated by (41b), the translation of (40b).

(41) a. $C_m$ $S_m$ S $O_m$ O $V_m$ V

b. *[*disambar ke mahine mein*]$_{Cm}$ [*amarikaa ke*]$_{Sm}$ [*raashtrapati*]$_S$ [*raajasthaan kee*]$_{Om}$ [*raajadhaani kee*]$_O$ [*sair karenge*]$_V$

DECEMBER-OF MONTH-IN AMERICA-OF PRESIDENT RAJASTHAN-OF CAPITAL-OF TOUR WILL-DO

The morphological variations are richer in Hindi than in English. The case markers *mein*, *se*, *ko*, *kaa* etc. are postpositioned and are strongly bound to the nouns. This allows Hindi to be a relatively free-word-order language. English uses PPs as complements and qualifiers, and the order of the words is quite fixed.

The free word ordering, however, poses difficulties in the analysis of the Hindi sentences. In addition to the phrase and clause attachment problems, it also makes the task of distinguishing the clauses and phrases from the subject and object of the sentence difficult, as they all have case markers and can be placed anywhere in the sentence. The sentences in (42) exemplify this.

(42) a. jeem ne choree karanevaale ladake ko laathee se maaraa
JIM-subj STEAL DO-ING-WHO BOY-TO STICK-WITH HIT

b. jeem ne laathee se choree karanevaale ladake ko maaraa
JIM-subj STICK-WITH STEAL DO-ING-WHO BOY-TO HIT

c. choree karanevaale ladake ko jeem ne laathee se maaraa
STEAL DO-ING-WHO BOY-TO JIM-subj STICK-WITH HIT

d. choree karnevaale ladake ko laathee se jeem ne maaraa
STEAL DO-ING-WHO BOY-TO STICK-WITH JIM-subj HIT

e laathee se jeem ne choree karnevaale ladake ko maaraa
STICK-WITH JIM-subj STEAL DO-ING-WHO BOY-TO HIT

'Jim hit with a stick the boy who had stolen.'

Here, *jeem ne* 'Jim' denotes the agent, *ladake ko* 'to the boy' the object and *laathee se* 'with a stick' the instrument. *Choree karanevaale* 'stealing'

is a clause qualifying *ladakaa*. Relative positions of each of these phrases can be varied as is apparent from the sentences in (42).

However, the postposition markers in Hindi always stay next to the nouns they modify and also have comparatively fixed roles. This partially compensates for the extra processing arising from the free word ordering.

English overloads the prepositions. For the UNL generation, not only PP attachment but also the semantic relation of the PP to the noun or the verb should be determined. Some examples are shown in (43).

(43) a. John ate rice with curd.
```
cob(eat(icl>do).@entry.@past, curd(icl>food))
```
b. John ate rice with a spoon.
```
ins(eat(icl>do).@entry.@past,
spoon(icl>thing).@indef)
```
c. John ate rice with Mary.
```
cag(eat(icl>do).@entry.@past, curd(icl>food))
```
d. The Demon ate the rice with the goat. `cob` or `cag`?

In the sentences in (43), the PPs starting with *with* have different roles. In (43a), the relation is co-object, in (43b) it is instrument and in (43c) it is co-agent. It is difficult to decide whether *goat* in (43d) is a co-object or a co-agent. The system identifies these relations using the semantic attributes of the nouns placed in the lexicon. This analysis is explained in detail in Parikh et al. (2000).

Hindi is a null-sSubject language, while English is not. Null-subject languages allow subjects to be dropped when the meaning is clear. **Error! Reference source not found.** above is an example of a Hindi sentence where the subject is dropped. Null-subject languages do not have pleonastics. This phenomenon is discussed in Section 6.1.4.

A very important feature of the Hindi language is that of conjunct and compound verbs which are formed by combining two or more verbs or by combining a noun or an adjective or an adverb with verbs like kr *kar* ‘do’or hao *ho* ‘be’. In the case of conjunct verbs, the first verb is usually the main one and the other is the subsidiary. All transformations of voice, mood, tense, person, gender and number affect the subsidiary verb only. The sentences in (44) exemplify this.

(44) a. vaha gaane lagee
SHE SINGING STARTED
‘She started singing.’
b. hma gaanao lagaoMgao.

*ham gaane lagenge*
WE SINGING WILL-START
'We shall start singing.'

The sentences in (45) show some of the interesting ways the verb जा *jaa* 'go' is used to emphasise or intensify the effects of the main verb. The literal translations show only the most common meanings of the constituent verbs.

(45) a. chale jaao
WALK GO
'Go away.'
b. ruk jaao
STOP GO
'Stop there.'
c. jhuk jaao
BEND GO
'Bend down.'

The phenomenon of compounding of verbs is a typical Indian-language phenomenon. The strategy to deal with this, however, is quite simple. The presence of two verbs next to each other provides the clue that the second verb is the intensifier, and generally the UNL expression produced gets the attribute `@emphasis` attached to the first verb as in (46) for example.

(46)
```
agt(go(icl>do).@entry.@imperative.@emphasis,
    you(icl>person))
```

There are numerous lexical and syntactic differences between Hindi and English, as described in the following sections.

## 5.1 NUMBER

Some words in English are always used in plural form, for example, *scissors*. This phenomenon does not occur in Hindi. It is impossible to determine from (47b) whether the reference is to one or more *scissors*.

(47) a. The company manufactures scissors. (many)
b. The scissors are very sharp. (one or many)

In Hindi, there are two different morphological forms for *scissors*: *kainchee* and *kainchiyaan* (plural), and thus this problem does not arise.

In English, some words have a single meaning in the singular form and multiple meanings in the plural. For example, the word *premise* means 'assumption', while the word *premises* mean 'assumptions' or 'the place that includes the building and the surrounding land'. Both these forms

should occur in the UW dictionary. This leads to the problem of the correct UW selection when the word *premises* occurs in a sentence. For these words, for example, the lexicon needs to store the UWs `premise(icl>assumption)` and `premises(icl>place)`. The question of choosing the right sense of *premise* as in *clean the premises* will, however, arise, and this can be resolved only by using the lexical properties of the main verb and the surrounding words.

Hindi, like Japanese, has a special way to show respect. It uses plural forms of pronoun for this purpose. For example, *aap* 'all of you' is used instead of *too* 'you' for a person when addressed with respect, and *ham* 'we' is used for *main* 'I' to show one's own importance. English does not have such practices. Thus while translating from English to Hindi, the sentence produced may be unacceptable for a native speaker of Hindi. For example, *too* used instead of *aap* addressing one's father or a distinguished person will be frowned upon. We have explained in Section 4.3 through the sentence **Error! Reference source not found.** the strategy for dealing with this phenomenon.

## 5.2 PERSON

The person of a noun does not generally change in translating between Hindi and English. But there is one situation where this occurs, and this happens more with spoken Hindi than with the written form. Hindi speakers often use the second person plural form instead of the third person singular to describe a person who is being interviewed or is in focus of an event. (48) shows an example.

(48) aap ne amarikaa se apanee p.h.d. kee upaadhi praapt kee
YOU-pl-subj AMERICA-FROM YOUR-pl PHD-OF DEGREE OBTAIN-ED
'He/She obtained his/her Ph.D. degree from America.'

It is not easy to deal with case. The fact that *aap* translates as *he/she* can be known only from the discourse and that currently the UNL handles only single sentences, calls for post editing of the UNL expressions.

## 5.3 GENDER

Three gender forms are recognized in English: masculine, feminine and neuter, while in Hindi there are only two forms: masculine and feminine. This does not pose much of a difficulty in translation from Hindi to English

and vice versa since the L-UW dictionaries are different for the two languages. The gender attributes are language dependent. For the UW `child(icl>human)`, the English mapping *child* has the neuter gender, while the Hindi mapping *bachcha* has the masculine gender.

The other differences with respect to the gender occur with pronouns and the possessive case. Hindi does not have different pronouns for different genders. For example, there are *he* and *she* in the third person in English, but there is only a single pronoun *vah* in Hindi. The verb morphology helps to identify the gender. The Hindi EnConverter by default generates *he* for [vah]. This mapping obviously is kept in the dictionary.

Gender-specific possessive pronouns (*his*, *her* or *its*) are used in English, while in Hindi, ]sa *us* is used for both the genders. On the other hand, Hindi expresses the gender of the possessed entity by using different case markers. For example, in Hindi, *usakaa dost* 'his/her (male) friend' or *usakee dost* 'his/her (female) friend' is used to refer to a boyfriend or a girlfriend respectively. In English the possessive preposition *of* is common for all genders, while in Hindi the corresponding case markers *kaa* (male) and *kee* (female) are used according to the gender of the possessed entity.

## 5.4 TENSE

There are irregular verbs in English which require separate entries in the dictionary, since the irregular verbs cannot be morphologically derived in a simple way from the stems. In Hindi also, there are irregular transformations of verbs. For example, *kar* 'do' and *kiyaa* 'did'. An important distinction in terms of the tense is that English does not show any inflection from the stem for the future tense, but uses auxiliaries like *will* and *shall* as in (49), while in Hindi, the present continuous tense does not show any inflection (50).

(49) He will read. He will write.

(50) vah padh rahaa hai. vah likh rahaa hai.
HE READING IS HE WRITING IS
'He is reading.' 'He is writing.'

Here, pZ *padh* 'read' and ilaK *likh* 'write' are the base morphemes for all possible transformations with respect to tense and person. These phenomena are dealt with through the elaborate set of morphology rules in the analyser.

## 6. Language Divergence between Hindi and English

We have already described the major differences between Hindi and English. In this section, we discuss them in a more formal setting proposed by Dorr (1993) which classifies various language divergences and suggests solutions to them with respect to LCS.

Unlike LCS, UNL is based on the linking of word concepts in a semantic net-like representation. We aim to show that most of the divergences described by Dorr either do not affect UNL-based translations or are comparatively easier to handle than in the LCS approach. Wherever possible, the examples from Dorr (1993) are used.

### 6.1 SYNTACTIC DIVERGENCE

Dorr gives the following divergences arising from structural and syntactic aspects of German, Spanish and English languages:

- Constituent order divergence
- Adjunction divergence
- Preposition-stranding divergence
- Movement divergence
- Null subject divergence
- Dative divergence
- Pleonastic divergence

In this section, we discuss the effect of each of these on the analysis of English and Hindi into the UNL form and also of generation from UNL into Hindi.

#### 6.1.1 *Constituent Order Divergence*

Constituent order divergence relates to the word-order distinctions between English and Hindi. Essentially, the constituent order describes where the specifier and the complements of a phrase are positioned. For example, in English the complement of a verb is placed after the verb and the specifier of the verb is placed before. Thus English is an SVO language. Hindi, on the other hand, is an SOV language. Example (51) shows the constituent order divergence between English and Hindi.

(51) Jim is playing tennis.

| S | V | O | |
|---|---|---|---|
| *jeem* | *Tenis* | *khel* | *rahaa hai* |

JIM TENNIS PLAYING IS
S O V

Also, in Hindi, the qualifier of the complement succeeds the verb whereas in English, it succeeds the complement, cf. (52).

(52) He saw a girl whose eyes were blue.
S V O Q
*us ne ek ladakee ko dekhaa jisakee aankhen neelee thee*
HE-subj ONE GIRL-TO SAW WHOSE EYES BLUE WERE
S O V Q

The UNL expressions generated from both English and Hindi are the same for these examples. In general, constituent order divergence does not affect the results of the EnConverter. But it does affect the strategy of analysis. The EnConverter system requires two UWs or compound UWs to be adjacent to each other to generate a UNL expression between them. After every relation is generated, one of the participating UWs is deleted from the node list and is made the child of the other UW in the semantic tree. For Hindi, the complement and its qualifier cannot be adjacent at any point of the analysis. Hence the SOV structure of the input sentence is converted in the intermediate steps into the SVO structure. The UNL expressions generated for the above example are shown in (53).

(53)
```
[S]
aoj(see(icl>do).@past.@pred.@entry, he(icl>person))
obj(see(icl>do).@past.@pred.@entry, girl(icl>person))
pof(girl(icl>person), eye(icl>thing).@pl)
aoj(blue(icl>state),eye((icl>thing).@pl)
[/S]
```

### 6.1.2 *Adjunction Divergence*

Syntactic divergences associated with different types of adjunct structures are classified as Adjunction divergence. Hindi and English differ in the possible positioning of the adjective phrase. In the former, this phrase can be placed to the left of the head noun. This is not allowed in English (54).

(54) * the [living in Delhi] boy
[*dillee mein rahanevaalaa*] *ladakaa*
DELHI-IN LIVING BOY

The suffix vaalaa *vaalaa* added to rhnaa *rahanaa* ‘live’ makes it an adjective phrase. This construction, in general, applies only to habitual actions. Consider the examples in (55).

(55) a. jeem ne *[*peetar ko pasand aanevaalaa*]* tohafaa bhejaa
JIM-subj [PETER-TO LIKE COMING] GIFT SENT

b. jeem ne vah tohafaa bhejaa jo peetar ko pasand aayaa
JIM-subj THAT GIFT SENT THAT PETER-TO LIKE CAME
c. jeem ne vah tohafaa bheejaa jo peetar ko pasand hai
JIM-subj THAT GIFT SENT THAT PETER-TO LIKE IS
'Jim sent the gift that Peter likes.'

Sentences (55a,c) are equivalent. (55b) cannot use *vaalaa*. The UNL expressions of the sentence (55) are shown in (56).

(56)
```
[S]
agt(send(icl>do).@entry.@past,Jim(icl>person))
obj(send(icl>do).@entry.@past,
  gift(icl>object).@indef)
aoj(like(icl>do).@present, Peter(icl>person))
obj(like(icl>do).@present, gift(icl>object).@indef)
[/S]
```

The generator identifies an adjective clause by the two arrows coming into the noun node `gift(icl>object)` from the verb nodes `send(icl>do)` and `like(icl>do)`. It identifies the main verb of the sentence by the `@entry` attribute. It generates the sentence (55a) if the verb `like(icl>do)` is in the present tense and the sentence (55b) if the verb is in the past tense.

Another divergence in this category is PP adjunction with respect to a verb phrase. In Hindi a PP can be placed between a verb and its object or before the object, while in English it can only be at the maximal level (i.e. not between the verb and its object (57–58).

(57) a. He called me [to his house].
b. * He called [to his house] me.

(58) a. usne mujhe *[*apne ghar*]* bulaayaa
HE TO-ME HIS HOUSE CALLED
b. usne *[*apne ghar*]* mujhe bulaayaa
HE HIS HOUSE TO-ME CALLED

The UNL expressions for both the sentences remain the same (59) and the generator can produce either of the Hindi sentences in (58).

(59)
```
[S]
agt(call(icl>do).@past.@pred.@entry, he(icl>person))
obj(call(icl>do).@past.@pred.@entry, I(icl>person))
plt(call(icl>do).@past.@pred.@entry,
  house(icl>place))
[/S]
```

### 6.1.3 *Preposition-stranding Divergence*

This divergence is accounted for by the choice of proper governors. Consider (60).

(60) a. Which shop did John go to?

b. * kis dukaan john gayaa mein
WHICH SHOP JOHN WENT TO

c.
```
[S]
agt(go(icl>do).@past,@pred.@entry,
   John(icl>person))
plt(go(icl>do).@past,@pred.@entry, shop(icl>place))
mod(shop(icl>place), which)
[/S]
```

Example (60b), which is a literal translation of (60a), is syntactically incorrect, as the case marker *mein* 'to' cannot be a proper governor for the noun phrase. In English, the preposition *to* is a proper governor for the trace. The case marker *mein* is required to follow the noun which in this case is *dukaan* 'shop'. The Hindi generator does the syntax planning accordingly and produces the right case marker when it encounters `plt(go(icl>do), shop(icl>place))`.

### 6.1.4 *Null-subject Divergence*

In Hindi, unlike in English, the subject of the sentence can be left implicit as in (61), for example.

(61) Long ago, there was a king.
*bahut pahale ek raajaa thaa*
LONG AGO ONE KING WAS

Hindi allows dropping of the subject where the subject is obvious as in **Error! Reference source not found.**, repeated here for convenience.

(31) jaarahaa hun
GOING AM
* 'am going.'

The subject *main* 'I' is absent. Such omissions are permitted only in two situations. The first is that a pleonastic is eliminated and the second is when a valid subject is omitted as its implicit presence is reflected through the morphology of the predicate. The first case is discussed in the next sub-section. In the other case, the eliminated subject must be produced in the UNL expressions. This is done by examining the structure of the UNL graph during the analysis. `aoj` and `agt` are the only relations that relate the predicate with the subject of the sentence. The system takes care of this

phenomenon by detecting the absence of the `agt` or `aoj` relation with the main predicate in a non-passive sentence. If such a condition is detected then it inserts an appropriate UW, `I(icl>person)` in example (31), in the nodelist. The analysis of the sentence is then continued as usual. The UNL representation for **Error! Reference source not found.** is shown in (62).

(62)
```
[S]
agt(go(icl>do).@entry.@present.@progress,
    I(icl>person))
[/S]
```

### 6.1.5 *Pleonastic Divergence*

A special kind of null-subject divergence is the Pleonastic Divergence. A pleonastic is a syntactic constituent that has no semantic content, as in (63), for example.

(63) It is raining.

*It* has no semantic role in (63). Similarly in sentence (61) above, *there* does not have any semantic role. Frequently, pleonastics are linked to another constituent that carries the appropriate semantic content. If the UNL representation of (63) is done as in (64a), then the Hindi generator will probably generate the sentence (64b), which is stylistically incorrect.

(64) a.
```
[S]
aoj(rain(icl>do).@progress.@entry,
    it(icl>abstract thing))
[/S]
```

b. ? `yah baareesh ho rahee hai`
THIS RAIN HAPPEN -ING IS

To deal with such problems, pleonastics are identified using semantic properties of the words in the sentence and they do not become part of the UNL expressions. For example, it has been observed that natural events like rain, thunder, snow, etc. make sentences using *it* as a pleonastic. Such words are given an attribute `NATURAL-EVENT` in the lexicon, using which, the *it* in the sentence, as in (63), is eliminated from the UNL expressions. Now, the UNL representation of (63) is (65).

(65)
```
[w]
rain(icl>do).@entry.@pred.@progress
[/w]
```

Note that the UW `rain(icl>do)` is not related to any other word and the event is described by a single UW which means ‘rain is in progress’. This can be translated to a correct form of (64) as (66).

(66) baareesh ho rahee hai
THIS RAIN HAPPEN -ING IS

Detailed information about detecting pleonastics can be found in Parikh and Bhattacharyya (2001).

## 6.2 LEXICAL-SEMANTIC DIVERGENCE

Lexical-semantic divergence (Dorr, 1993) arising from the properties of the entries in the lexicon is of the following types:

- Conflational divergence
- Structural divergence
- Categorial divergence
- Head swapping divergence
- Lexical divergence

These are explained with examples along with their effect on the analyser and generator outputs.

### 6.2.1 *Conflational Divergence*

“Conflation” is the lexical incorporation of necessary components of meaning (or arguments) of a given action. This divergence arises from a variation in the selection of the word between the source language and the target language, as in (67) for example.

(67) a. Jim stabbed John.
b. jeem ne john ko chhoore-se maaraa
JIM-subj JOHN-TO KNIFE-WITH HIT

Here, *stab* does not have a single-word equivalent word in Hindi. We require the phrase CUro sao maara *chhoore se maaraa* ‘hit with a knife’. As a result, the UNL expressions generated from (67a) and (67b) vary. The Hindi analyser produces (68).

(68)
```
[S]
agt(hit(icl>do).@entry, Jim(icl>person))
ben(hit(icl>do).@entry, John(icl>person))
ins(hit(icl>do).@entry, knife(icl>thing))
[/S]
```

However, the English analyser directly produces `stab(icl>do)`. But if the Hindi phrase *chhoore se maaraa* is mapped to the UW `stab(icl>do)` in the Hindi–UW dictionary, the Hindi analyser produces (69).

(69)
```
[S]
agt(stab(icl>do).@entry, Jim(icl>person))
ben(stab(icl>do).@entry, John(icl>person))
[/S]
```

The EnConverter's property of picking up the longest lexeme has been exploited here. The expression is the same as the UNL expressions produced by the English analyser. Most cases of conflational divergence are handled this way. The opposite case of Hindi words being conflational has been discussed in Section 2.2 for both noun (*devar*) and verb (*ausaanaa*).

### 6.2.2 *Structural Divergence*

Consider example (70).

(70) Jim entered the house.
*Jeem ne ghar mein pravesha kiyaa*
JIM-subj HOUSE-INTO ENTRY DID

The Hindi sentence diverges structurally from the English sentence, since the verbal object is realized as a noun phrase (*house*) in English and as a prepositional phrase (*ghar mein* 'into the house') in Hindi. In English, both *enter* and *enter into* will be allowed whereas in Hindi the prepositional phrase should strictly be used. The UNL expressions from both the English and Hindi sentences are the same (71).

(71)
```
[S]
agt(enter(icl>do).@entry.@pred.@past,
    Jim(icl>person))
plt(enter(icl>do).@entry.@pred.@past,
    house(icl>place))
[/S]
```

If *into* is not present, the English analyser can generate `obj` between *enter* and *house*. This problem is solved by using the semantic attribute `PLACE` of the word *house* in the lexicon. This causes the generation of `plt` instead of `obj`. Thus, the lack of syntactic information (implicit prepositions) is compensated for by the semantic knowledge.

### 6.2.3 *Categorial Divergence*

Categorial divergence arises if the lexical category of a word changes during the translation process. Consider (72).

(72) They are competing.
*vaha muqaabalaa kar rahe hai*
THEY COMPETITION DOING ARE

Here, *competing* is expressed as a verb in English and as a noun–verb combination (*muqaabalaa kar* 'do competition') in Hindi. This divergence is very common in English-to-Hindi MT, and in general in English to an Indian language MT. Hindi, like most Indian languages, forms "combination verbs" in which a noun is followed by a form of *kar* 'do' or *ho* 'be' to express the action suggested by the noun.

This phenomenon is handled by the Hindi analyser by having two entries for such nouns in the lexicon: one as a noun and the other as a verb. The verb entry has an attribute link that indicates that a form of *kar* is to follow the noun. For the example in (72), *muqaablaa* has the two entries in the lexicon shown in (73).

(73) [*muqaablaa*] `{} "competition(icl>action) " (N, NA, MALE, INANI, ABSTRACT);`
`[muqaablaa] {} "compete(icl>do)" (V, link);`

Because of this, the UNL expressions for both the English and the Hindi sentences are the same (74).

(74)
```
[S]
agt(compete(icl>do).@entry.@pred.@present.@progress,
    they(icl>person))
[/S]
```

### 6.2.4 *Head-swapping Divergence*

Head-swapping divergence divides between demotional and promotional divergence. Demotional divergence is characterized by the demotion (placement into a position lower down) of a logical head. In such a situation, the logical head is associated with the syntactic adjunct position and then the logical argument is associated with a syntactic head position.

For example, in (75), the word *suffice* is realized as the main verb in English but as an adjectival modifier *kaafee hai* in Hindi.

(75) a. It suffices.
b. yaha kaafee hai
IT SUFFICIENT IS

The UNL expressions generated from the English and Hindi analysers differ. The English analyser generates (76a) while the Hindi analyser generates (76b).

(76) a.
```
[S]
aoj(suffice(icl>do).@entry.@present, it)
[/S]
```
b.
```
[S]
aoj(sufficient.@entry.@present, it)
[/S]
```

The Hindi generator produces the sentence (75b) from both these representations. This is because the Hindi–UW dictionary has `suffice(icl>do)` mapped to *kaafee hai* 'is sufficient'. Hindi does not have any equivalent verb for *suffice*. Thus the divergence is handled in the lexicon with the entry (77).

(77) [*kaafee*] `{} "suffice(icl>do)" (V, VI);`

Promotional divergence is characterized by the promotion (placement into a higher position) of a logical modifier. The logical modifier is associated with the syntactic head position and then the logical head is associated with an internal argument position, as exemplified in (78).

(78) The play is on.

*khel chal rahaa hai*

PLAY GO -ING IS

Here the modifier *is on* is realized as an adverbial phrase in English but as the main verb *chal rahaa hai* 'is going on' in Hindi. The UNL expressions generated by the English and Hindi analysers are shown in (79).

(79) a.
```
[S]
aoj(on(icl>state).@entry.@present,
  play(icl>abstract thing).@def)
[/S]
```
b.
```
[S]
aoj(go on(icl>occur).@entry.@present.@progress,
  play(icl>abstract thing))
[/S]
```

The solution to this is same as that for demotional divergence. The dictionary entry in this case would be (80).

(80) [cala] `{} "go on" (V,Va);`

### 6.2.5 *Lexical Divergence*

Lexical divergence means that the choice of a target-language word is not a literal translation of the source-language word. However, lexical divergence arises only in the context of other divergence types. In particular, lexical divergence generally co-occurs with conflational, structural and categorial divergences. An example is shown in (81).

(81) john jabarjasti ghar mein ghus gayaa
JOHN FORCEFULLY HOUSE-IN ENTER WENT
'John broke into the house.'

Here the divergence is lexical in the sense that the target-language word is not a literal translation of the source-language word. The English and Hindi analysers will both produce the UNL expressions in (82).

(82)
```
[S]
agt(enter(icl>do).@past.@entry.@force,
    John(icl>person))
plc(enter(icl>do).@past.@entry.@force,
    house(icl>home))
[/S]
```

It is clear how the Hindi analyser can produce the above expressions. The English analyser achieves this by mapping *break into* to `enter(icl>do)` in the English–UW dictionary. It also places an attribute `FORCED` into the lexicon which signals the generation of `@force` during analysis.

## 7. Experimental Observations

The English Analyser, the Hindi Analyser and the Hindi Generator have been tested using the sentences in the United Nations Charter provided by the UNU. The corpus was designed to test the DeConverters of different languages all over the world. The corpus has around 180 sentences. It is in English and has been manually translated into Hindi for the Hindi analyser. As the analysers are not yet equipped with word-sense disambiguation capability, inter-category word senses were manually disambiguated. As mentioned before, the analysers have intra-category or part-of-speech disambiguation capability. Approximately 80% of these sentences have been successfully converted to UNL expressions by the analysers without any change in the input sentences. The rest had to be pre-edited to a certain extent by simplifying the structure of the sentences and controlling the use of punctuation. The UNL expressions generated by the English and Hindi analysers were given to the Hindi generator. 95% of these UNL expressions were correctly converted into Hindi by the Hindi generator.

The Hindi analyser has also been tested on a huge Hindi corpus provided by the Ministry of Information Technology, Government of India. This corpus consisted mainly of stories from the political domain. The English analyser too has been tested on documents like the EnConverter Manual, sentences from Brown corpus and stock-market stories downloaded from different web sites. We are continuously upgrading our system by testing on numerous corpora. The test base is currently considerable. The "Barcelona corpus" obtained from the multilingual information-processing research being conducted in Spain, sentences from the *Medline* corpus, agricultural corpora from the Gujarat Government and other such corpora are being worked on. Thus the evaluation process is in progress.

Besides techno-scientific domains we have tested the analyser on literary works also. It is worth noting here that such sentences require more pre-processing than sentences from the technical domains. An example of a sentence not handled properly by the system is (83), taken from a P.G. Wodehouse novel.

(83) I loosed it down the hatch, and after undergoing the passing discomfort, unavoidable when you drink Jeeves's patent morning revivers, of having the top of the skull fly up to the ceiling and the eyes shoot out of their sockets and rebound from the opposite wall like racquet balls, felt better.

However, with some obvious pre-editing as shown in (84), the sentence is analysed accurately.

(84) I loosed it down the hatch and after undergoing the passing discomfort which is unavoidable when you drink Jeeves's patent morning revivers, of having that the top of the skull fly up to the ceiling and the eyes shoot out of their sockets and rebound from the opposite wall like racquet balls, felt better

The verification of the analysis and generation processes have been carried out by converting Hindi sentences into UNL expressions and generating the sentence back. The results obtained are quite satisfactory in the sense that the generated sentences are in most cases the same as the source sentences. Sometimes the postposition markers are different while at other places a different word has been chosen. Yet other times, the structure of the generated sentence differs from the source sentence. However, in all cases the idea contained in the source sentence is conveyed

in the generated sentence. Some examples are shown in (85)–(87). In each case, (a) shows the source sentence, (b) the UNL expression, and (c) the sentence generated. Differences between (a) and (c) are highlighted.

(85) a. adhyayana samooha upakaran aur sevaaoM se saMbandhit bahoot saare muddoM ko samaavisht karate haiM.
'The Study Groups cover a wide number of issues related to equipments and services.'

b.
```
[S]
aoj(cover(icl>include):21.@entry.@present.@pred,
   Study Groups:00)
obj(cover(icl>include):21.@entry.@present.@pred,
   issue(icl>important point):1R.@pl)
mod(issue(icl>important point):1R.@pl,
   relate(icl>concerning):14)
mod(issue(icl>important point):1R.@pl,wide number
   of(icl>very great):1F.@pl)
aoj(relate(icl>concerning):14,:01)
and:01(service(icl>assistance):0U.@entry.@pl,
   equipment(icl>tool):0G)
[/S]
```

c. *adhyayana samooha upakaran aur sevaaoM saMbandhit bahoot saare muddoM ko samaavisht karate haiM.*

Comparing (85a,c) we find that only the postposition marker of saovaaoM*sevaa* 'service' has changed. The sentence is acceptable in Hindi and the meaning of course is conveyed.

(86) a. antarraashtriiya saMsthaa ke roop meM aaii tii yoo sarakaaroM aur gair-sarakaarii saMsthaoM ko doorasaMchaar taMtra aur sevaaoM ke paricaalan ke vistaar aur samanvayiikaraN hetu kaarya karane ke lie aur sabhii deshoM tak unakii pahuMch ko baDAvA dene ke lie eka saath laataa hai.
'As an international organization, ITU brings together governments and private sectors to work for expanding and coordinating the operation of the telecommunication networks and services, and to promote their access to all countries.'

b.
```
[S]
aoj(bring
   together(icl>gather):6T.@entry.@present.@pred,
ITU(icl>International Telecommunication Union):0X)
obj(bring
   together(icl>gather):6T.@entry.@present.@pred,
   :01)
pur(bring
   together(icl>gather):6T.@entry.@present.@pred,
   :04)
```

```
and:04(foster(icl>nurture):69.@entry.@pred,
  work(icl>do work):4J.@pred)
obj:04(foster(icl>nurture):69.@entry.@pred,
  access(icl>approach):5X)
scn:04(access(icl>approach):5X,  country(icl>
  nation):5G.@pl)
mod:04(access(icl>approach):5X,
  those(icl>pronoun):5R)
aoj:04(overall(icl>all):5B,
  country(icl>nation):5G.@pl)
pur:04(work(icl>do work):4J.@pred,     :03)
mod:03(coordination(icl>coordinating):3Y.@entry,
  operation(icl>functioning):38)
and:03(coordination(icl>coordinating):3Y.@entry,
  expanding(icl>expansion):3M)
mod:03(operation(icl>functioning):38,  :02)
mod:03(:02, telecommunication:2B)
and:02(service(icl>assistance):2Y.@entry.@pl,
  network(icl>system):2N)
and:01(institution(icl>organization):1Z.@entry.@pl,
  government:16.@pl)
aoj:01(private(ant>governmental):1K,
  institution(icl>organization):1Z.@entry.@pl)
aoj(ITU(icl>International Telecommunication
  Union):0X,  as:0L)
obj(as:0L, institution(icl>organization):0E)
aoj(international(icl>characteristic):00,
  institution(icl>organization):0E)
[/S]
```

c. sarakaaroM aur gair-sarakaarii saMsthaoM ko antarraashtriiya saMsthaa ke roop meM aaii tii yoo doorasaMchaar kii taMtra aur sevaaeM paricaalan ke vistaar aur samanvayiikaraN ke lie kaarya karane aur sabhii deshoM meM unakii pahuMch ko baDAvA dene ke lie eka saath laataa hai.

Here the phrase *sarakaaroM aur gair-sarakaarii saMsthaoM ko* 'governments and private sectors' has been placed at the start of the sentence. Being followed by *ke roop meM* 'as' this gives an impression initially that *ITU* is being qualified by the phrase. This, however, gets rectified as one reads ahead. The meaning is conveyed, but the source sentence is structurally better than the generated one. There are other minor changes like *sevaaoM* becoming *sevaaeM* (these are two different plural forms of *sevaa* meaning *service* and usable interchangeably) and tk *tak*

(preposition *to*) becoming *meM* (preposition *in*), which do not alter the meaning much.

(87) a. yah utsav pradarshanoM kaa ek badaa kaaryakram aur saaMskritik kriyaa-kalaapoM kaa eka vistrit kshetra pradaan karegaa jo poore 155 dinoM tak vishva saMskrtiyoM kii srjanaatmakataa par dhyaan kendriwt karegaa.

'This Festival will offer a broad programme of performances and a wide range of cultural activities that will focus on the creativity of world cultures over a period of 155 days.'

b.
```
[S]
obj(provide(icl>do):2Q.@entry.@future.@pred, :01)
aoj(provide(icl>do):2Q.@entry.@future.@pred,
   festival(icl>event):05)
mod(festival(icl>event):05, this:00)
aoj(focus(icl>concentrate):4W.@future.@pred, :01)
and:01(range(icl>variety):2B.@entry,
   program(icl>performance):10)
mod:01(range(icl>variety):2B.@entry,
   activity(icl>action):1Q.@pl)
aoj:01(cultural(aoj>thing):1F,
   activity(icl>action):1Q.@pl)
mod:01(program(icl>performance):10,
   performance(icl>abstract thing):0E.@pl)
aoj:01(great(icl>characteristic):0U,
   program(icl>performance):10)
tim(focus(icl>concentrate):4W.@future.@pred,
   day(icl>period):3H.@pl)
scn(focus(icl>concentrate):4W.@future.@pred,
   creativity(icl>creativeness):4D)
mod(creativity(icl>creativeness):4D,
   culture(icl>civilisation):3Y.@pl)
aoj(world(mod<thing):3S,
   culture(icl>civilisation):3Y.@pl)
aoj(around(icl>about):38, day(icl>period):3H.@pl)
qua(day(icl>period):3H.@pl, 155:3D)
[/S]
```

c. *yaha utsav saaMskritik kriyaa-kalaapoM ke badaa pradarshanoM kaa eka kaaryakram aur eka vibhinn prakaar pradaan karegaa jo poore 155 dinoM meM vishva saMskrtiyoM kii racanAwmakawaa par dhyaan kendriwt karegaa.*

This illustrates changes of word as in (i) *vibhinn prakaar* 'range' in place of *vistrit kshetra* (another meaning of 'range') (ii) *racanAwmakawaa* in place of *srjanaatmakataa* both meaning the same, i.e. 'creativity' and (iii) *meM*

'in' in place of *tak* 'over'. The reordering of phrases, however, is more serious as in *saaMskritik kriyaa-kalaapoM ke badaa pradarshanoM kaa eka kaaryakram* ('a programme of a broad performance of cultural activities') replacing *pradarshanoM kaa ek badaa kaaryakram aur saaMskritik kriyaa-kalaapoM* 'a broad programme of performances and cultural activities' where meaning alteration within that part of the sentence has taken place. The generated sentence, however, is not far in meaning from the source sentence.

Example (88) shows that though sentences in English and Hindi with identical meaning are represented as different sets of UNL expressions by the English and Hindi analysers, the Hindi generator generates the same output for both the representations. (88b) shows the UNL expressions generated by the English analyser. The same sentence was manually translated to Hindi (88c) and input to the Hindi analyser, the output of which was (88d). The output of the Hindi generator for both (88b,d) is (88e).

(88) a. UNEP has a mission to care for the environment.

b.
```
[S]
aoj(have(icl>state):05.@entry.@present,
   UNEP(icl>United Nations Environment
   Programme):00)
obj(have(icl>state):05.@entry.@present,
   mission(icl>duty):0B.@indef)
pur(care(icl>do):0M.@present.@pred,
   environment(icl>state):0Z.@def)
pur(mission(icl>duty):0B.@indef,
   care(icl>do):0M.@present.@pred)
[/S]
```

c. U N E P kaa lakshya paryaavaran kee dekhabhaal karnaa hai
UNEP-OF MISSION ENVIRONMENT-OF CARE DO IS

d.
```
[S]
obj(care(icl>do):1I.@entry.@present.@pred,
   environment(icl>abstract thing):13)
mod(mission(icl>duty):0W, UNEP(icl>United Nations
   Environment Programme):0H)
aoj(care(icl>do):1I.@entry.@present.@pred,
   mission(icl>duty):0W)
[/S]
```

e. U N E P kaa lakshya paryaavaran kaa khyaal rakhnaa hai
UNEP-OF MISSION ENVIRONMENT-OF CARE DO IS

'UNEP has a mission to care for the environment.'

This lends credence to the capturing of the semantics by the UNL is a language-independent way.

At this stage, it is difficult to compare the computational complexity of the analysis of Hindi and English sentences into UNL. However, we mention a few pointers in that direction:

1. UNL is based on a predicate-centric framework. The analyser needs to know the predicate before it starts generating the UNL expressions. Because of the SOV structure of Hindi, in most cases, the verb occurs at the end of the sentence. Thus the Hindi analyser has to do a complete morphological analysis of the words on its way to the end of the sentence. There are examples in which the Hindi analyser completes the morphological analysis of words until the end of the sentence and then comes all the way back to the subject of the sentence. This normally does not happen in the case of the English analyser. As soon as it encounters the predicate, it can start dealing with the complements and the PPs.

   The SOV structure also causes problems because of the computational model adopted. For example, the adjacency requirement of the logical units or constituents described in Section 6.1.1, sometimes calls for manipulations like the exchange of syntactic constituents to change their order in the sentence.
2. Prepositions in English can be proper governors (Dorr, 1993). Thus sentences like (89a) need to be dealt with. The system is required to produce (89b).

   (89) a. Which shop did John go to?

   b. `plt(go(icl>do).@entry.@interrogation.@past, shop(icl>place))`

   But because of the computational model adopted *to* is required to be adjacent to *shop*. This is achieved by exchanging *go* and *shop* when they are adjacent to each other in the node-list. Such computations can become very complex in the case of longer sentences with long-distance dependencies. In Hindi the case markers cling to the noun they govern leading to simpler computation.
3. The problem of word-sense disambiguation poses difficulties for both the analysers. UNL requires the analysers to generate an unambiguous word concept. Neither the English nor the Hindi analyser has any support for sense disambiguation. However, both perform very well for part-of-speech disambiguation. This helps prune options for a UW.

4. Our experiments show that the number of rules fired is nearly the same for both English and Hindi analysis in most cases. This number is directly proportional to the number of lexemes. At least two rules, "shift" and "process", are required for each morpheme. Hindi generally requires more morphological analysis. Thus the number of rules fired is a bit more than that of English. To illustrate this, the statistics for four sentences (90)–(93) are given in Table III.

(90) UNIFEM works to promote the economic and political empowerment of women.
*yunifem ouraton ke aarthik tathaa raajanaitik adhikaar ko badhaavaa dene ke liye kaarya karatee hai.*
UNIFEM WOMEN-OF ECONOMIC AND POLITICAL EMPOWERMENT-TO PROMOTE-GIVE-FOR WORK-DOING-IS

(91) I know the lady who has worn a blue saree.
*mai us ourat ko jaanataa hun jisane neelee saadee pahanee hai.*
I THAT WOMAN-TO KNOW-AM WHO BLUE SAREE HAS-WORN

(92) Uncle told us that Gita is removing dust from the kitchen with a broom.
*chaachaa ne ham se kahaa ki geetaa rasoighar mein jhaadoo se dhool nikaal rahee hai*
UNCLE US-TO TOLD THAT GITA KITCHEN-IN BROOM-WITH DUST REMOVING IS

(93) With Lord Krips, his wife had also come and she wanted to buy a fine shawl from India for taking home.
*lord krips ke saath unakee patnee bhee aaee huee thee or ve bhaarat se svadesh le jaane ke liye ek umdaa shaal khareedanaa chaahatee thee.*
LORD KRIPS-WITH HIS WIFE ALSO COME-HAD AND SHE INDIA-FROM NATIVE-LAND TAKE-GO-FOR ONE FINE SHAWL BUY WANT -ED

| Example | Type | No. of lexemes | | No. of rules fired | |
|---|---|---|---|---|---|
| | | Engl. | Hindi | English | Hindi |
| 90 | Simple | 22 | 30 | 54 | 64 |

| | | | | | |
|---|---|---|---|---|---|
| 91 | Adjective Clause | 20 | 20 | 46 | 55 |
| 92 | Noun Clause | 26 | 33 | 57 | 71 |
| 93 | Compound | 44 | 55 | 101 | 122 |

*Table III*: Statistical information for example sentences (90)–(93)

The difference in the number of rules fired can be accounted for from the fact of two rules used per lexeme. The other contributing factors are:

a. Simple (90): The presence of the conjunction in the sentence. English requires looking ahead by several words to make sure it is not a compound sentence and is a simple conjunction of nouns. The morphology of Hindi helps in avoiding this processing.
b. Adjective clause (91): The adjective clause requires the Hindi analyser to do extra processing as explained in Section 6.1.1. This explains the nine extra rules fired by the Hindi analyser.
c. Noun clause (92): The difference here is exactly proportional to the difference in the number of morphemes.
d. Compound (93): An extra rule fires in the case of the English analyser. This is for the look-ahead processing of the compound sentence.

## 8. The Issue of Disambiguation

As has been mentioned at various places in this paper, our system currently does mainly part-of-speech disambiguation and a little bit of sense disambiguation for postposition markers and *wh*-pronouns. The main instruments of disambiguation are the condition windows around the analysis heads and also the lexical attributes of the words. This achieves the look ahead and look back necessary for disambiguation. We point out the specific example sentences mentioned in the paper where disambiguation takes place.

In (19), with four uses of the word *desert*, part-of-speech disambiguation uses the adverb *totally* that must precede an adjective that in turn must precede a noun. In (21a), disambiguation of *when* used to indicate an adverb phrase (… *to my home when* ...) uses the fact that *home* does not have "time" attribute, while in (21b), disambiguation of the adjective phrase (… *a time when* …) depends on the fact that *when* can qualify a noun with a "time" attribute.

In the four sentences with *se* in Hindi (26)–(29), sense disambiguation uses the lexical attributes of the preceding nouns, as it does with the sentences using *with* in (43).

These examples throw light on the disambiguation capability of the analysers. However, more powerful lexical resources will have to be used for large-scale word-sense disambiguation.

## 9. Conclusions and Future Directions

The criteria for deciding the effectiveness of an interlingua are that (a) the meaning conveyed by the source text should be apparent from the interlingual representation and (b) a generator should be able to produce a target-language sentence that a native speaker of that language accepts as natural. A careful observer will notice that (a) and (b) are essentially the same. Still we put them down separately to emphasize the presence of a mechanical procedure in (b).

Keeping these criteria in view, our conclusions on the capability of the UNL vis-à-vis language divergence especially between English and Hindi are:

1. The UNL expressions generated from English and Hindi texts are mostly the same, as has been brought out in Section 6.
2. When they differ, they do so mainly in the case of very overloaded constructs like *have* where the mechanical analyser does not capture the varied nuances.
3. The lexical-semantic divergence is actually handled in the L-UW dictionary. The generator primarily bears the burden of naturalness and idiomaticity in this case.
4. The syntactic divergence, on the other hand, is primarily tackled by the analysers. The capability is built into the rules.
5. The amenability to generation is being tested through at least another language, Marathi, a western Indian language, in our case. The results are approximately the same as in Hindi because of the similarity in structure between Hindi and Marathi.

There are several future directions. The L-UW dictionary has to be enriched enormously both in terms of the UW content and the semantic attributes so as to capture the word and world knowledge. The analysers need to be augmented with powerful word-sense disambiguation modules.

The Hindi generator needs to be thoroughly tested using the UNL expressions produced by the analysers for other languages. Investigation of the UNL as a knowledge representation scheme and the use of this knowledge for various purposes like text summarisation, automatic hypertext linking, document classification, text-image consistency checking and such other knowledge-intensive tasks should be carried out.

## Notes

[1] NI indicates that the noun ends with an *i* (Romanised Hindi). This information helps in morphological analysis.

[2] EnConverter and DeConverter are tools provided by the UNL Project, Institute for Advanced Studies, United Nations University, Tokyo (UNU, 2000).

## References

Arnold, D. and L. des Tombes: 1987, 'Basic theory and methodology in EUROTRA' in Nirenburg (1987), pp. 114–135.

Arnold, Doug and Louisa Sadler: 1990, 'Theoretical Basis of MiMo', *Machine Translation* **5**:195–222.

Bhandari V.: 2002, 'Machine Translation of Phrasal Verbs from English to Hindi', *Bachelor of Technology Project Report*, Department of Computer Science and Engineering, Indian Institute of Technology, Kanpur, India.

Bharati, Akshar, Vineet Chaitanya and Rajeev Sangal: 1995, *Natural Language Processing: A Paninian Perspective*, New Delhi: Prentice Hall of India.

Boitet, Christian: 1988, 'Pros and Cons of the Pivot and Transfer Approaches in Multilingual Machine Translation,' in Maxwell et al. (1988), pp. 93–106.

Carbonell, Jaime G. and Masaru Tomita: 1987, 'Knowledge Based Machine Translation, the CMU Approach,' In Nirenburg (1987), pp. 68–89.

Dorr, Bonnie J.: 1992, 'The use of lexical semantics in interlingual machine translation,' *Machine Translation* **7**: 135–193.

Dorr, Bonnie J.: 1993, *Machine Translation: a View from the Lexicon*, The MIT Press, Cambridge, Mass.

D'Souza, Rayner, G.V. Shivakumar, D. Swathi and Pushpak Bhattacharyya: 2001, 'Natural Language Generation from Semantic Net like Structures with application to Hindi,' in *STRANS–2001: Symposium on Translation Support Systems*, Kanpur, India.

Gopinathan, S. and S. Kandaswamy (eds): 1993, Anauvaad kI samasyaaEM [Problems of Translation], Lokbharti Prakashan, New Delhi, India.

Guha, R.V., D.B. Lenat, K. Pittman, D. Pratt, and M. Shepherd: 1990, 'Cyc: A Midterm Report,' *AI Magazine*, pp. 32-59.

Hardt, S.L.: 1987, 'Primitives' in S.C. Shapiro (ed.), *Encyclopedia of Artificial Intelligence*, John Wiley and Sons, New York, NY, pp. 759-761.

Katz, J.: 1966, *The Philosophy of Language*, Harper and Row, New York, pp 156.

M. King (ed.): 1987, *Machine Translation Today: The State of the Art*, Edinburgh University Press, Edinburgh.

King, M. and S. Perschke: 1987, 'EUROTRA', in King (1987), pp. 373–391.

Senniappan, Kumarvel and Pushpak Bhattacharyya: 2000, 'Automatic Generation of Hyperlinks using Semantic Information,' in *International Conference on Information Technology (CIT 2000)*, Bhubaneshwar, India, pp. 43-48.

Landsbergen, Jan.: 1987, 'Isomorphic grammars and their use in the ROSETTA translation system.' In King (1987), pp. 351–372 .

Lytinen, Steven and Roger C. Schank: 1982, 'Representation and Translation,' Technical Report, Department of Computer Science, Yale University, New Haven, CT.

Martin, John C.: 1991, *Introduction to Languages and the Theory of Computation*, McGraw Hill, New York, NY.

Maxwell, Dan, Klaus Schubert and Toon Witkam (eds): 1988, *New Directions in Machine Translation*, Foris, Dordrecht.

Monju, M., T. Shilpa, D. Smita, Leena Giri, Shachi Dave and Pushpak Bhattacharyya: 2000, 'Knowledge Extraction from Hindi Text', In *Proceedings of the International Conference on Knowledge Based Computer Systems (KBCS 2000)*, Mumbai, India, pp. 193-204.

Muraki, Kazunori: 1987, 'PIVOT: Two-Phase Machine Translation System,' in *MT Summit Manuscripts and Program*, Hakone, Japan, pp. 81–83.

Nirenburg, Sergei (ed.): 1987, *Machine Translation: Theoretical and Methodological Issues*, Cambridge University Press, Cambridge.

Nirenburg, Sergei, Jaime Carbonell, Masaru Tomita and Kenneth Goodman: 1992, *Machine Translation: A Knowledge Based Approach*, Morgan Kaufmann, San Mateo, CA.

Nirenburg, Sergei, Victor Raskin and Allen B. Tucker: 1987, 'The Sturcture of Interlingua in TRANSLATOR,' in Nirenburg (1987), pp. 90–113.

Okumura, Akitoshi, Kazunori Muraki and Susume Akamine: 1991, 'Multi-lingual Sentence Generation from the PIVOT Interlingua.' In *MT Summit III*, Washington, DC, pp. 67–71.

Parikh, Jignashu and Pushpak Bhattacharyya: 2001, 'Towards Realising *The Semantic Web*' submitted to *Natural Language Engineering*.

Parikh, Jignashu, Trilok Soni and Chirag Shah: 2000, 'Conversion of English Language Texts to Universal Networking Language,' *Bachelor of Engineering Dissertaion*, Dharamsinh Desai Institute of Technology, Nadiad, India.

Perschke, Sergei: 1989, 'EUROTRA', in *MT Summit II*, Munich, pp. 113–119.

Rao, Durgesh, Kavita Mohanraj, Jayprakash Hegde, Vivek Mehta and Parag Mahadane: 2000, 'A Practical Framework for Syntactic Transfer of Compound-Complex Sentences for English-Hindi Machine Translation', in *International Conference on Knowledge Based Computer Systems (KBCS 2000)*, Mumbai, India, pp. 343-354.

Schank, Roger C.: 1972, 'Conceptual Dependency: A Theory of Natural Language Understanding,' *Cognitive Psychology* **3**:552–631.

Schank, Roger C.: 1973, 'Identification of Conceptualizations Underlying Natural Language', in Roger C. Schank and K. M. Colby (eds), *Computer Models of Thought and Language*, Freeman, San Francisco, CA, pp. 187–247.

Schank, Roger C. (ed.): 1975, *Conceptual Information Processing*, Elsevier Science Publishers, Amsterdam.

Schank, Roger C. and Robert Abelson: 1977, *Scripts, Plans, Goals and Understanding*, Lawrence Erlbaum, Hillsdate, NJ.

Schubert, K.: 1988, 'The architectre of DLT – Interlingual or Double Direct?' in Maxwell et al. (1988), pp. 131–143.

Schütz, Jörg, Gregor Thurmair and Roberto Cencioni: 1991, 'An Architecture Sketch of Eurotra-II', in *MT Summit III*, Washington, DC, pp. 3–10.

Tiwari Bholanath and Naresh Kumar: 1987, Problems of translation from various foreign languages], Prabhat Prakashan, New Delhi, India.

Uchida, Hiroshi: 1989, 'ATLAS', In *MT Summit II*, Munich, pp. 152–157.

UNL: 1998, The Universal Networking Language (UNL) specifications version 3.0. Technical Report, United Nations University, Tokyo.

UNU: 2000a, *DeConverter Specification Version 2.0*, UNU/IAS/UNL Center, Tokyo.

UNU: 2000b, *EnConverter Specification Version 2.1*, UNU/IAS/UNL Center, Tokyo.

Vauquois, Bernard and Christian Boitet: 1985, 'Automated Translation at Grenoble University,' *Computational Linguistics* **11**:28–36.

Vauquois, Bernard: 1975, *La traduction automatique à Grenoble*, Dunod, Paris.

Wahlster, W.: 1993, 'Verbmobil: Translation of Face-to-Face Dialogs', in *The Fourth machine Translation Summit: MT Summit IV*, Kobe, Japan, pp. 127–135.

Wilks, Yorick: 1972, *Grammar, Meaning and the Machine Analysis of Language*, Routledge & Kegan Paul, London.

Wilks, Yorick: 1987, 'Primitives', in S.C. Shapiro (ed.), *Encyclopedia of Artificial Intelligence*, John Wiley and Sons, New York, NY, pp. 759–761.

Witkam, T.: 1988, 'DLT – An industrial R&D project for multilingual MT', in *COLING Budapest: Proceedings of the 12th International Conference on Computational Linguistics*, Budapest, pp. 756–759.

Woods WA: 1985, What's in a link: Foundations for semantic networks. In Brachman RJ, Levesque HJ, editors, Readings in Knowledge Representation 218--241, Morgan Kaufmann Publishers, Inc., San Mateo, CA.