ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology

Vol. 1, Issue 1, November 2012

| A Monthly Peer Reviewed & Referred Journal || Impact Factor: 1.672|

Natural Language Analysis and Generation in the Framework of Universal Networking Language

Vijaya R Varma Pothuri[1], Rama C Koppula[2], Venkateswarlu Chennareddy[3]

Senior Consultant, Capgemini Consulting, Bangalore, India[1]

Senior Staff Software Engineer, IBM India Software Labs, Hyderabad, India[2]

Senior Staff Software Engineer, IBM India Software Labs, Hyderabad, India[3]

ABSTRACT: Machine Translation (MT) has evolved from basic word-to-word translation to systems requiring deep linguistic, contextual, and subject-specific knowledge. Early methods failed to address the complexities of language, but advances have improved accuracy and fluency. With the rise of the internet, demand for multilingual content translation has surged, as language barriers restrict access to vast online information. This paper examines the challenges and developments in MT, highlighting the Universal Networking Language (UNL) project, which aims to create a global communication platform. By improving cross-linguistic communication, MT can enhance global information access and understanding.

KEYWORDS: Natural Language, NLP, UNL, Artificial Intelligence

I. INTRODUCTION

Machine Translation is about translation of a sentence in one natural language to another natural language using a machine. In early days, total conversion was thought to be impossible. The popular misconception was Machine Translation is a waste of time because a machine can never be made to translate the words of Shakespeare.

Initially it was thought that a system which contained the information that exists in a standard bilingual dictionary was the process that was considered. Later some rules to change the order of worlds in the input sentence were developed so that the order is more characteristic of the target language. Some languages like Hindi take the object before the web therefore for the translation of the sentence He eats banana, a rule is generated to order it into He banana eats.

The simple idea of translation of words and using reordering rules did not give good results. It has been found that in addition to a good knowledge of vocabulary of both the source and the natural language, the grammar of both the languages (the system which specifies which sentences are well formed in a language and which are not), knowledge of the nature of things out in the world, and technical knowledge of the text in subject area are also needed [4]. Most of the efforts of late have been concentrated on increasing the breadth and depth of the linguistic or grammatical knowledge available in the system.

The Internet has emerged as the global infrastructure, revolutionizing access to information, as well as the speech by which it is transmitted and received. With the technology of electronic mail, for example, people may communicate rapidly over long distances. Not all users, however, can use their own language for communication. The smoothest way to communicate with other people and obtain information and education, is in one's mother language. Smooth communication among people with different languages will improve mutual understanding. The Universal Networking Language (UNL) is developed for this purpose. UNL will provide a common communication environment for different languages.

1.1 Motivation

The rapid proliferation of the Internet has resulted in a number of documents available to the people via browser programs. These documents vary in their content from news and sports, to history, economics and philosophy. The variety of documents available is huge and the quality of information is enormous. With the rise in the number of users of the Internet, the information flow on the Internet has also risen. The barrier which obstructs the smooth transfer of information is language. Documents are available in different languages. Documents available in natural language may not be understood by some people. To make the facility of natural languages which does the job automatically.

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology

Vol. 1, Issue 1, November 2012

| A Monthly Peer Reviewed & Referred Journal || Impact Factor: 1.672|

II. NATURAL LANGUAGE ANALYSIS AND GENERATION

2.1 Introduction

Natural Language Processing deals with making the machine understand textual information presented to it and the ability to converse with a human being. Natural Language Processing includes both analysis or parsing and generation. The main problem involved in doing the whole taks is the ambiguity involved in natural languages. The entire natural language processing problem can be divided into two main [6] tasks.

- Processing written text, using lexical, syntactic and semantic knowledge of the language which requires real world information.
- Processing spoken language, using all the information needed above plus additional knowledge about phonology, as well as enough added information to handle the further ambiguities that arise in speech.

2.2 The phases in Natural Language Analysis

The process of Natural Language Analysis involves the following phases.

2.2.1 Morphological Analysis

In this phase individual works are analyzed, non-word tokens like periods , commas, and other punctuation marks are eliminated. From the sentence A cup of tea., the words a, cup, of, tea are obtained. The suffixes and prefixes of the worlds involved are stripped off. The syntactic categories of all the worlds are assigned in this phase.

2.2.2 Syntactic Analysis

The process of inferring structure from grammatical strings is called parsing. This plays an important role in Natural Language Analysis for the following two reasons [3].

- Semantic processing must operate on sentence constituents. Parsing helps by constraining the constituents that semantics can order.
- In many cases extracting the meaning of a sentence requires the grammatical information about it. Parsing the English sentence helps in getting the syntactic constructs.

2.2.3 Semantic Analysis

The main function of semantic analysis is development of contents and finding out the relationships between them. In this phase the structures created by the syntactic phase are assigned meanings. In other worlds, a mapping is made between the syntactic structures and the objects in the task domain. The structures used to represent the knowledge includes Semantic nets, Frames and Scripts. From the syntactic structures obtained in the previous phase, a mapping is done onto these grammars for semantic nets, If ta mapping cannot be properly done then the sentence that is considered termed as semantically incorrect the sentence He is dead and alive., is an example of semantically ill formed sentence, though not when used figuratively. These types of sentences are rejected in the semantic analysis phase.

2.2.4 Knowledge Representation using semantic nets

Semantic nets [6] are a way of representing knowledge. The main idea behind semantic nets is that the meaning of a concept comes from the ways in which it is connected to other concepts. In a semantic net, information is represented as a set of nodes connected to each other by a set of labeled arcs, which represent relations among nodes.

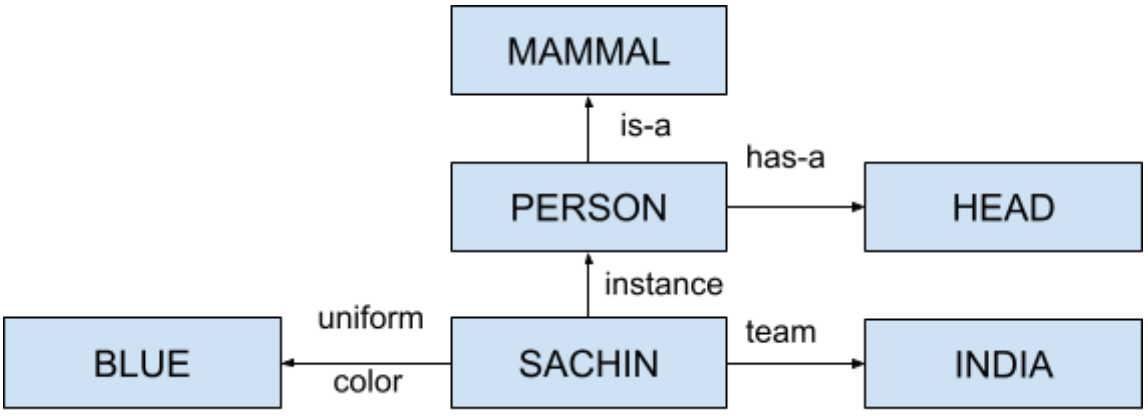


Figure 2.1: Example of Semantic Network

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology

Vol. 1, Issue 1, November 2012

| A Monthly Peer Reviewed & Referred Journal || Impact Factor: 1.672|

One of the early ways that semantic nets were used was to find relationships among objects by spreading activation out from each of the nodes and seeing where the activations meet. This process is called Intersection Search. Using intersection search and the network above, the questions like what is the connection between India and Blue? Can be answered from figure 2.1.

Advantages of entity based organization of knowledge, that slot and filler representations provide, is taken for answering the above questions. To answer more structured questions, however, required networks that are more structured, Semantic nets are a natural way to represent relationships that appear as a ground instance of binary predicates in predicate logic.
e.g,
isa(person, mammal)
instance(Gandhi,person)
The semantic nets are not only for binary predicates, predicates of any arity can be represented using semantic nets.
Unary predicates like **man**(Gandhi) can be converted to **instance**(Gandhi, man).

Three or more palace predicates can also be converted to a binary form by creating one new object representing the entire predicate statement and then introducing binary predicates to describe the relationship to this new object of the original arguments.

2.2.5 Discourse Integration and Programmatic Analysis
To understand a single sentence, the contents of the previous sentence need to be known. The reference to objects or human beings by pronouns can be analyzed only by the information given by the previous statements. This phase does that analysis, Rajive Gandhi was Prime Minister of India for 6 years. He was killed in the year 1991 by the Tamil militants. The 'he' in the second statement refers to Rajive Gandhi. This fact can be known when the first statement is also considered.

2.3 Ambiguities in Parsing
2.3.1 Lexical Ambiguity
There are some words in the English language which can be used as different parts of speech. For example, the words like run, drink, book, fly etc. can be used as both verbs and nouns. The decision of classifying whether a word is a noun or a verb depends on the place of it in the sentence. Various methods [3] are used to obtain heuristics and to isolate which sense of a word is being implied by the word in a particular sentence.

2.3.2 Prepositional Phrase Ambiguity
For sentences with a prepositional clause, it is hard to identify which word does it modify. The possibilities are the main verb, the direct object,and the indirect object. If the sentence I saw the boy with telescope. is considered, there are two parses for it. One indicating that the prepositional phrase with the telescope is attached to the verb swa, and the other indicating that it is attached to the direct object boy.

2.4 natural language Generation
Natural Language Generation (NLG) is the subfield of artificial intelligence and computational linguistics that is concerned with the construction of computer systems that can produce understandable texts in English or other human languages from some underlying non-linguistic representation of information [5].

Natural language generation systems combine knowledge about language and the application domain to automatically produce documents, reports, explanations, help messages, and other kinds of texts.

2.5 Applications for Natural Language Generation
The most common use of natural language generation technology is to create computer systems that present information to people in a representation that they find easy to comprehend. Internally, computer systems use representations which are straightforward for them to manipulate, such as databases, accounting spreadsheets, expert system knowledge bases, simulations of physical systems etc.

2.6 NLG Tasks
The task of a natural language generation system can be characterized as mapping from some input data to an output text. As this is a computational process, it is necessary to decompose this task into a number of more finely characterized sub-steps[5].

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology

Vol. 1, Issue 1, November 2012

| A Monthly Peer Reviewed & Referred Journal || Impact Factor: 1.672|

2.6.1 Content Determination

Content Determination is the process of deciding that information should be communicated in the text. This process can be described as creating a set of MESSAGES from the system's inputs or underlying data sources. These messages are the data objects then used by the subsequent language generation process.

Generally, the message creation process largely consists of filtering and summarizing input data, and the messages created are expressed in some formal languages that labels and distinguishes what we might think of as the entities, concepts and relations in the domain. If we consider a rail travel information system, specific trains, places and times can be viewed as entities, the property of being the next train as a concept, and the notions such as departure and arrival are relations between trained and times. Examples of Messages are shown in Figure 2.2.

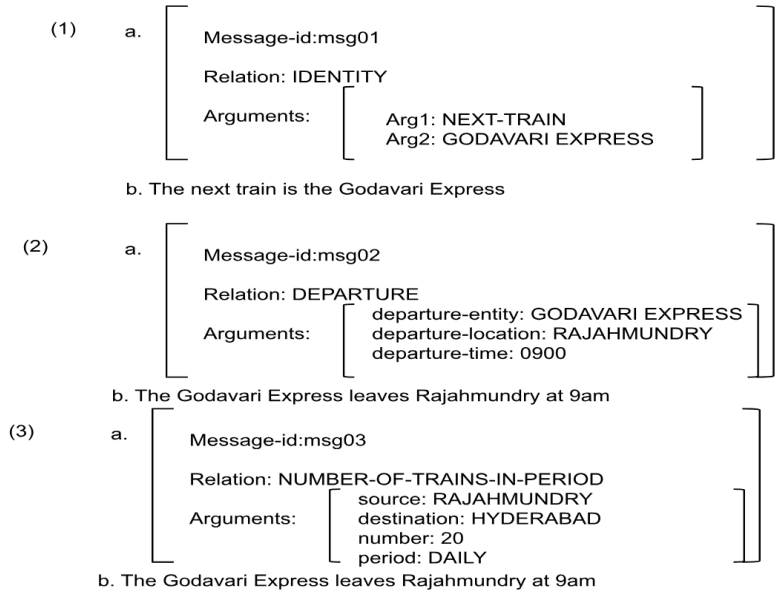


Figure 2.2 Examples of Messages

2.6.2 Discourse planning

Discourse planning is the process of imposing ordering and structure over the set of messages to be conveyed. Good structuring can make a text much easier to read. The result of the Discourse planning is usually represented as a tree structure. The leaf nodes of the tree specify individual messages, and the internal nodes specify how messages are grouped together and related to each other. The clustering decisions made in the tree will have an impact on the determination of sentence and paragraph boundaries in the resulting text.

In the rail travel information system, the NUMBER-OF-TRAINS-IN-PERIOD and IDENTITY messages are placed in a SEQUENCE relationship, and the DEPARTURE message is taken to be and ELABORATION of the IDENTITY message. The tree structure is given in Figure 2.1.

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology

Vol. 1, Issue 1, November 2012

| A Monthly Peer Reviewed & Referred Journal || Impact Factor: 1.672|

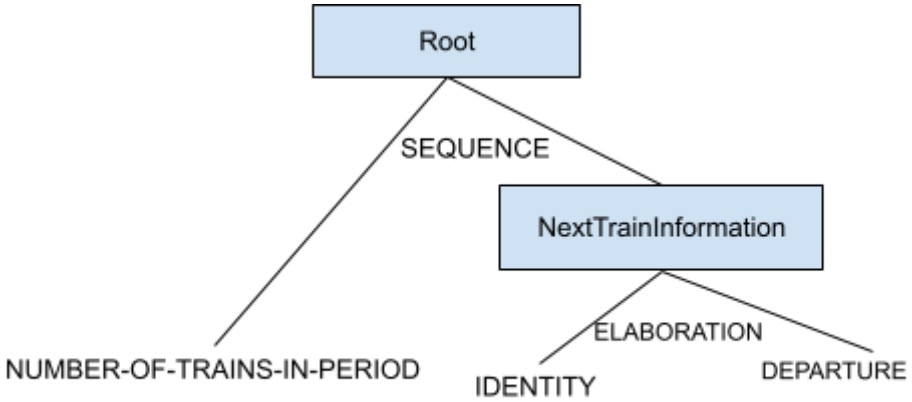


Figure 2.3: Tree Structure for Discourse Planning

2.6.3 Sentence Aggregation

Sentence Aggregation is the process of grouping messages together into sentences. Aggregation is not always necessary-each message can be expressed in a separate sentence but in many cases good aggregation can significantly enhance the fluency and readability of a text. In principle, aggregation techniques can be used to form paragraphs and other higher-order structures as well as sentences. In our example, sentence aggregation could combine the IDENTITY and DEPARTURE messages into a single sentence, which would be related as The next train, which leaves at 9am, is the godavari express.

2.6.4 Lexicalization

Lexicalization is the process of deciding which specific words and phrases should be closed to express domain concepts and relations which appear in the messages. In many cases lexicalization can be done trivially by hard-coding a specific word or phrase for each domain concept or relation.

For the event DEPARTURE, the words leave and depart are both possibilities. For example, we might simply specify that the DEPARTER message should always be expressed by the word leave. In some cases, fluency can be improved by allowing the NLG system to vary the words used to express a concept or relation. For example, departing is more formal than leaving.

2.6.5 Referring Expression Generation

Referring Expression Generation is the task of selecting words or phrases to identify domain entities. Referring expression generation is closely related to lexicalization, since it is also concerned with producing surface linguistic forms which identify domain elements. Unlike lexicalization, referring expression generation is usually formalized as a discrimination task, where the system needs to communicate sufficient information to distinguish one domain entity from other domain entities.

2.6.6 Linguistic Realization

Linguistic Realization is the process of applying the rules of grammar to produce a text which is syntactically, morphologically, and orthographically correct or it is defined as the task of generating correct sentences to communicate messages. For example, a linguistic realization process may decide to express the NUMBER-OF-TRAINS-IN-PERIOD message above as the following sentence.

There are 20 trains each day from Rajahmundry to Hyderabad

In this example, the syntactic component of the realizer has decided to add the function words from and to to mark those parts of the sentence which specify the train's source and destination; the morphological component has produced the plural form trains of the root word train; and the orthographic component has capitalized the first word of the sentence and added full stop at the end of the sentence.

2.7 NLG Architectures

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology

Vol. 1, Issue 1, November 2012

| A Monthly Peer Reviewed & Referred Journal || Impact Factor: 1.672|

There are many ways of building a system that performs the tasks we have just described. The simplest approach is to build a separate module for each task, and connect these modules via a one-way pipeline [5]. In such an architecture, the content determination module first decides on all the messages to be included in the text; the discourse planning module then organizes these messages into a discourse structure tree; and so on.

The most common architecture in present-day NLG systems is a three-stage pipeline with the following stages [5]

Text planning: This stage combines content determination and discourse planning tasks. In many real applications, it is difficult to separate these two tasks.

Sentence Planning: This stage combines sentence aggregation, lexicalization, and referring expression generation.

Linguistic Realization: This task involves syntactic, morphological, and orthographic processing.

III. UNIVERSAL NETWORKING LANGUAGE

The Universal Networking Language (UNL) is an artificial language in the form of a semantic network for computers to express and exchange every kind of information. It has been introduced as a digital meta language for describing, summarizing, refining, storing and disseminating information in a machine-independent and human-language-neutral form [7].

The motivation behind UNL is to develop an interlingua representation such that semantically equivalent sentences of all languages have the same interlingua representation. It is an application of the semantic net knowledge representation schema for machine translation. UNL plays the role of an interface between different languages to exchange information. The two main modules in the UNL are Enconverter and Deconverter. Enconverter converts the text from natural language into UNL expressions while the Deconverter converts from UNL expressions into natural language.

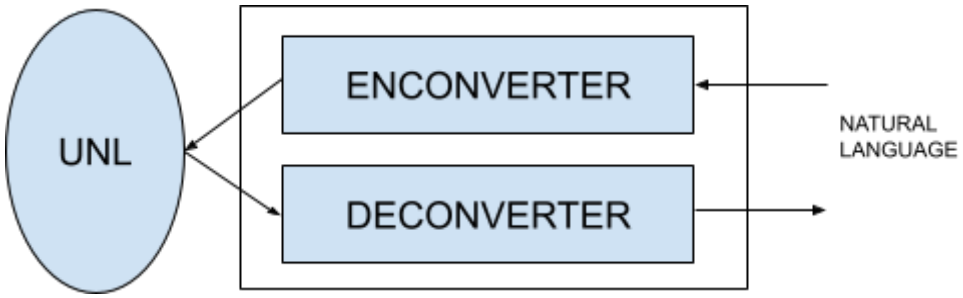


Figure 3.1: UNL System

3.1 Aims of UNL

UNL is a designed with the following aims:

1. It is capable of exactly representing the information expressed in any language.
2. Its expression must be as general as possible inorder to be understood by the people who are engaged in the development of Encovnerters and Deconverters in each language [7].

3.2 System Architecture of UNL

UNL system is a set of interrelated modules for the extraction, storage, retrieval and expression of information.

3.2.1 Extraction of Information

This is carried out by the module called Enconverter which translates a text from teh source language into UNL document.

3.2.2 Storage and processing of Information

Storage of Information is in the form of UNL documents. So, the information represented in the UNL will be in a machine and natural language independent form.

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology

Vol. 1, Issue 1, November 2012

| A Monthly Peer Reviewed & Referred Journal || Impact Factor: 1.672|

3.2.3 Retrieval of Information

Inorder to retrieve any information, the search engine will search for UNL expressions rather than searching for natural language character strings. UNL serves as an interface between the document base and search engine and the result of search or retrieval operation is a UNL document.

3.2.4 Expression of Information

Deconverter is the module which converts the UNL documents into a text in the language that Decovmerteer is intended for. The same UNL document can simultaneously be routed to different users for viewing in their respective languages depending upon the Deconverters.

3.3 UNL's representation of Sentence

UNL represents information i.e. meaning, sentence by sentence. Sentence information is represented as a semantic network with hyper-nodes. UNL expressions are unambiguous. In the UNL semantic network, nodes represent concepts, and arcs represent relations between concepts. UNL is composed of words expressing concepts called Universal Words(UWs) which are interlinked with other UWs to form sentences. These links, known as relations, specify the role of each word in a sentence. UNL expresses information classifying objectivity and subjectivity. Objectivity is expressed using UWs and relations. Subjectivity is expressed using attributes by attaching them to UWs.
UNL vocabulary consists of
-Universal Words (UWs).
Relation Labels.
Attribute Labels.

3.4 Universal Words

A Universal Word (UW) is a word in English that is applied as a label to denote a specific meaning i.e. Universal Words are made up of English words followed by a list of constraints and a list of attributes.

Syntax of Universal Word < UW >::=< HeadWord > [< ConstraintList >]

< HeadWord >::=< character >...

< ConstraintList >::= "(" < Constraint > ["," < Constraint >]...")"

< Constraint >::=< RelationLabel > " >"|" <" < UW >

< RelationLabel >::= "agt"|"and"|"aoj"|"obj"|"icl"|...

< AttributeList >::=< Attribute > ["." < Attribute >]...

< InstanceID >::=< Digit >< Digit >

< Attribute >::= "@" < AttributeLabel >

< Digit >::=0|1|2|...|9

< character >::= "a"|...|"z"|"A"|...|"Z"|"-"

Head Word Head Word is an English word that is interpreted as a label for a set of concepts.
Head word serves to organize concepts and make it easier to remember which is which.

Constraints or Restrictions The Constraint list restricts the interpretation of a UW to a subset or to a specific concept included within the Elementary UW.

Attributes The Constraint list is followed by a list of attributes, which provide the information about how the concept is being used in a particular sentence. Attribute Labels are quantifiers for a UW and are used to describe what is said from the speaker's point of view. These express sentential information such as tense, aspect, intention of utterance and sentence structure.
e.g.,

Attribute Label	Universal Word	Meaning
@pl	book(icl>document).@pl	books
@past	book(obj>place).@past	booked

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology

Vol. 1, Issue 1, November 2012

| A Monthly Peer Reviewed & Referred Journal || Impact Factor: 1.672|

Instance ID A UW can also include an instance ID. The instance ID is used to indicate some referential information that there are two different occurrences of the same concept. If the same UW occurs more than once, it is in all cases understood to refer to the same entity or occurrence.

e.g.,
Once Computer is interconnected with another Computer
Here the UW Computer has occurred twice. So, here these two words must be distinguished by using two instance ID's.
Computer(icl > machine) : 01
Computer(icl > machine) : 02
This makes it clear that the first computer is not connected to itself.

3.4.1 Three types of Universal Words

These words are listed inorder of practical importance:

Restricted UW's These are head words followed by a constraint list. The restrictions means "restrict your attention to this particular sense of word".
e.g., book(icl > document)
book(icl > event)
The first UW represents a more specific sense of a book that contains a set of pages. The later one represents a more specific sense of "reserving".

Extra UW's Extra UW's denote concepts that are not found in English and that have to be introduced as extra categories. Foreign language labels are used as head words.
e.g., kuchipudi(icl > dance)and
dosa(icl > food)

Elementary UW's These are character strings that correspond to an English word. These are bare head words with no constraint list.
e.g., go, take, house etc.

3.5 Relations

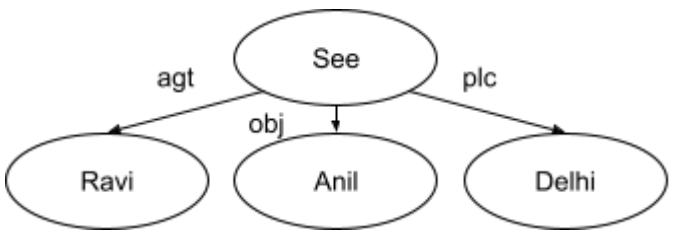
Binary Relations are the building blocks of UNL sentences. They are made up of a relation and two UW's. The relations between UW's have different labels according to the different roles they play. A relation label is represented as strings of three characters or less. A binary relation between UW's is defined as

rel(uw1,uw2)

Where rel is the relation label and uw1, uw2 are Universal words.

Syntax < *Binary Relation* > *::==* < *RelationLabel* > *[":"*< *compounduw - id* >*]*
<uw1> |":" <compundvw - id>","<uw2> |" : "< *compoundvw - id* >*)"*

There are 41 relations in the UNL system[10]. Some relations are explained below wit han example.
agt: defines a thing which initiates an action.
obj: defines a thing which is affected by an event or an action.
plc: describes the place where an event occurs or a state is true or a thing exists.
ex:
Ravi saw Anil in Delhi.
agt(see(icl>do(obj>thing)).@entry.@past.@complete,Ravi(icl>person))
obj(see(icl>do(obj>thing)).@entry.@past.@complete,Anil(icl>person))
plc(see(icl>do(obj>thing)).@entry.@past.@complete,Delhi(icl>place))



3.6 Compound Universal Words

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology

Vol. 1, Issue 1, November 2012

| A Monthly Peer Reviewed & Referred Journal || Impact Factor: 1.672|

Compound Universal Words are a set of Binary relations that are grouped together so that we can talk about them as a single unit.

Compound UW-ID's consist of digits (":"followed by two digits) and these are used to represent compound UW's. Attribute list may optionally follow a compound UW.
E.g.,
Arun travelled from Delhi to Chennai to see Anil.
agt(see.@pred,:01)
obj(see.@pred,Anil)
agt:01(travel.@pred.@entry.@post,Arun)
plf:01(travel.@pred.@entry.@post,Delhi)
plt:01(travel.@pred.@entry.@post,Chennai)
Here :01 is the Compound UW-ID.
Syntax <compounduw - id>::=":"<digit><digit>["."<AttributeList>]
<AttributeList>::=<Attribute>["."<Attribute>]
<Attribute>::="@" <AttributeLabel>
<AttributeLabel>::="reason"|"volitional"|"past"|...
<digit>::=0|1|...|9

Compound UW's are defined by placing a compound UW-ID immediately after the relation label in all of the binary relations that are grouped together.
Compound UW's can be cited or referred by simply using the compound UW-ID as an UW.

3.7 UNL System

The UNL system allows people to communicate with peoples of different languages in their mother tongue. The UNL system basically consists of language servers, UNL editors and UNL viewers [7].
A conversation system from native language into UNL is called "EnConverter", and one that deconverts from UNL into native language is called "DeConverter". Information "enconverted", from any language is exchanged in UNL format via networks. Information represented in UNL is "deconverted" into each native language on the terminal network.

3.7.1 Language Server

A Language Server consists of a deconverter and enconverter. The process of "enconversion" and "deconversion" are provided by a Language Server which resides in the network of the Internet. An enconverter is a software that enconverts natural language sentences into UNL. UNL center has developed software for enconversion called "Enco". "Enco" is language independent software applicable for any language. This "Enco" together with a language UW dictionary and conversion rules can convert a natural language sentence into UNL.

A Deconverter is a software that converts UNL expression into natural language. UNL center has developed a software for deconversion called "Deco". It is also language independent software that can be applied to any language. This "Deco" constitutes a deconverter together with language dictionary and Deconversion rules. The enconversion rules are used to analyze natural language sentences and deconversion rules are used to generate natural language sentences.

3.7.2 Language Dictionary

It is a language dependent component. It links each UW to a natural language word. It also consists of the morphological, syntactic and semantic attributes of the word. The same dictionary is used for encovnersion and deconversion.

3.7.3 UNLEditor and Viewer

UNL editor is used to make UNL documents. UNL editor is linked to a language server equipped with an "enconverter" and a "decovnerter" for natural language. As an author writes a document, e-mail or any other text, in his/her language, UNL editor "enconverts" it into UNL documents. In this process, UNL expressions are produced automatically or interactively with the author.

UNL viewer is used to see UNL documents in a user's native language; UNL viewer utilizes a language server when it deconverts the UNL documents into the user's native language.

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology

Vol. 1, Issue 1, November 2012

| A Monthly Peer Reviewed & Referred Journal || Impact Factor: 1.672|

IV. ENCONVERTER AND DECONVERTER

4.1 EnConverter

EnConverter is a language independent parser, which provides synchronously a framework for morphological, syntactic and semantic analysis. EnConverter inputs a string of sentences. The input string is scanned from left to right. Then all matched morphemes with the same starting characters are retrieved from the World Dictionary and become candidate morphemes. The rules are applied to these candidate morphemes according to a rule priority in order to build the syntactic tree and the semantic network of UNL for the sentence. The output of the whole process is a semantic network expressed in the UNL format [2].

4.1.1 Structure of EnConverter

EnConverter operates on the nodes of the Node-list through its windows. There are two types of windows, namely "Analysis Windows" circumscribed by the windows called "Condition Windows".

EnConverter analyzes a sentence using the Word Dictionary, Knowledge Base, and Enconversion Rules. It retrieves relevant dictionary entries from the Word Dictionary, operates on the nodes in the Node-list by applying Enconversion Rules, and generates semantic networks of UNL by consulting Knowledge Base [2].

EnConverter uses the Condition Windows to check the neighboring nodes on both sides of the Analysis Windows in order to determine whether the neighboring nodes satisfy the conditions for applying an encovnersion rule or not. The Analysis Windows are used to check two adjacent nodes in order to apply one of the enconversion rules[2]. If there is an applicable rule, EnConverter will modify the grammatical attributes of these two nodes, and/or create a partial syntactic tree and/or UNL network, according to the type of the rule. This process will be continued until all the necessary UNL networks are generated for the input sentence.

Word Dictionary The word entries of each language are stored in the Word Dictionary.
Each entry in the Word Dictionary is composed of three kinds of elements. The Head-word, the Universal Word (UW) and the Grammatical Attributes.

Enconversion Rule An Enconversion Rule is composed of COnditions for the nodes placed on the Analysis Windows and Condition Windows, and Actions and/or Operations for the nodes placed on the Analysis Windows.

Node-list The Node-list shows the current list of nodes that the EnConverter can look at through its windows.

Node-net The Node-net shows the current UNL expression network of output.

Knowledge Base All possible relations between each pair of UWs are defined in the UNL Knowledge Base(KB).

4.1.2 Function of EnConverter

EnConverter converts enconversion rules from text format into binary format. Next, EnConverter can input either a string or a list of morphemes|words for sentences of the native language, which are enclosed between Sentence Headnode([<<]) and Sentence Tailnode([>>]). EnCovnerter treats the morphemes between sentence headnode and sentence tailnodes as one input sentence.

When a list of morphemes is imputed, the nodes for all the morphemes are created in the Node-list as soon as it is retrieved from the Word Dictionary, and EnConverter works on the nodes without Word Dictionary Retrieval. Enconversion of the sentences of a native language to UNL is achieved by applying enconversion rules. EnConverter starts to apply rules to Node-list through its windows. Morphological, syntactic and semantic analysis are carried out synchronously in enconversion rule application. The process of rule application is to find a rule and to take actions or operate on the Node-list inorder to create a syntactic tree and UNL network using the nodes in the Analysis Windows.

If a word satisfies the conditions required for the window of a rule, thai word is selected and the rule application succeeds and this node is output into the Node-net. When the applicable rule cannot be found or rule application fails then backtracking occurs. When backtracking occurs, the process backtracks to the previous stage when the last analysis rule was applied, or a word/morphee was selected, and tries another possibility. This process will be continued

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology

Vol. 1, Issue 1, November 2012

| A Monthly Peer Reviewed & Referred Journal || Impact Factor: 1.672|

until the syntactic tree and UNL network are completed and only the entry node remains in the Node-list. Enconversion process stops when either the Sentence Tail moves to the left Analysis Window or the Sentence Head moved to the right Analysis Window. At the end only Entry nodes remain in the Node-list.

4.1.3 Output of EnConverter

The output of EnConverter is a set of UNL expressions composed of the UWs of the words (morphemes) that constitute the input text. If a sentence is converted perfectly, only the entry node (root node of the Node-net) will remain in the final state of the Nodel-list between the Sentence Head and Sentence Tail nodes. It can happen that more than two nodes still remain in the Node-list when the rule application ends because of cancellation or time over of the process, or an incomplete set of rules. In this case, EnConverter will output all the partial UNL expressions for each remaining node in the Node-list in the correct order.

4.2 DeConverter

DeConverter is a language independent generator that provides synchronously a framework for morphological and syntactic generation, and word selection for natural collection. DeConverter can deconvert UNL expressions into a variety of natural languages, using a different set of files such as the Word Dictionary, Grammatical Rules and Co-occurrence Dictionary of each language.

DeConverter transforms the sentence represented by an UNL expression, i.e. a set of binary relations, into the directed hyper-graph structure called Node-net. The root node of the Node-net is called Entry Node and represents the main precipitate of the sentence. Then it applies generation rules to every node in the Node-net respectively, and generates the word list in the target language. In this process, the syntactic structure is determined by applying Syntactic Rules, while morphemes are generated by applying Morphological Rules [1].

4.2.1 Structure of DeConverter

DeConverter operates on the nodes of the Node-list, and inserts nodes from Node-net into the Node-list through its windows. There are two types of windows, they are “Generation Windows”(GW) circumscribed by windows called “Condition Windows”(CW). DeConverter generates a sentence using the Word Dictionary, Decovnersion Rules and Co-occurrence Dictionary. It retrieves relevant dictionary entries from the Word Dictionary, operates or inserts nodes by applying Deconversion Rules.

The system applies generation rules according to the information in the Node-list and inserts each node of the node-net into the Node-list and then replaces each node in the Node-list with the corresponding word to get the final results. This process is based on the concept of Generating Windows (GW) and Condition Windows (CE). The latter are used to check the neighboring nodes on both sides of generation rule. The generation window is used to check the applicability of one of the generation rules and take the necessary action.

4.2.2 Function of DeConverter

DeConverter converts deconversion rules from text format into binary format. It inputs a sentence of UNL expressions and converts it into the Node-net. For each sentence, DeConverter first converts the UNL expressions into the input form for Deconverter, a directed hyper-graph called Node-net, and this entry node is placed on the Node-list of the initial state. A node of the Node-net is made corresponding to a UW of the input of UNL. Word Dictionary Retrieval is carried out using an UW as a search key. As a result of the Word Dictionary Retrieval, a list of word entries is linked to each node as candidates. Then, it applies decovnersion rules to the Node-list and inserts nodes from the Node-net. This process will end when either the Sentence Tail node of the Node-list appears in the left Generation Window or the Sentence head node appears in the right Generation Window.

There are two kinds of rules in deconversion, generation rules and post-editing rules. Generation rules describe how various UNL expressions should be expressed in a native language. By using the generation rules, the decovnersion of UNL expressions to a native language can be completed basically. Post-editing rules provide a number of convenient functions for editing the generated results.

4.2.3 Output of DeConverter

DeConverter outputs the target sentence when the deconversion of a UNL expression ends successfully. If the deconversion fails, no sentence is output. The output of Deconverter is a sentence in target language. If a UNL expression is deconverted perfectly, only the entry node (root node of the Node-net) will remain in the final state of the Node-list between the Sentence Head and Sentence Tail nodes.

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology

Vol. 1, Issue 1, November 2012

| A Monthly Peer Reviewed & Referred Journal || Impact Factor: 1.672|

V. CONCLUSION

UNL System is positioned as one the components of the information infrastructure that can be used for smoothe international exchange of information. A multi-language information platform that used UNL would not only enable communication using a user's mother tongue to many other languages. It would also remove the language barriers in every area of the information technology world. If the information from different languages was converted into the UNL information technology world. If information from different languages was converted into UNL is various software treating language, and in information resources such as information bases in various software treating language, and in information resources such as information bases and databases, the software and information resources could be shared throughout the world.

UNL currently provides 41 relation labels (RLs) representing the core semantic relations that hold between the universal words (UWs) that mek up the abstract UNL sentence representation. In order to be implemented as software tools, which can be incorporated into WWW browsers, UNL must have a sufficiently wide scope to address any subject or field domain. Accordingly, the necessary increase in scope and flexibility of the UNL representation language should be the subject of future investigations, which will need to consider both the language-related details to be included and the computational effort required to process the additional semantic concepts.

REFERENCES

[1] DeConverter and EnConverter Specification Version 2.1. UNU/IAS, UNL Center, Tokyo, 150-8304, Japan, 2000.
[2] James Allen, Natural Language Understanding. The Benjamin/Cummings Publishing Company, 1987 edition , 1987.
[3] D. J. Arnold, Lorna Baikan, Siety Meijer, R.Lee Humphreyas, and Louisa Sadler. Machine Translation Introductory Guide. 1995. http://www.essex.ac.uk/linguistics/clnt/MTbook.
[4] Ehud Reiter and Robert Dale,. Building applied natural language generation systems. Natural Language Engineering, 1997.
[5] Elaine Rich and Kevin Knight. Artificial Intelligence. Tata McGRAW-HILL edition, 2001.
[6] Zhu M., Uchida H. The universal networking language (UNL) specifications version 3.0 technical report, united nations university, tokyo. 1998.http://www.unl.unu.edu/unisys/unl/unis30.doc