

UNL2005 for Providing Knowledge Infrastructure

Hiroshi Uchida
UNL Center
UNDL Foundation
53-70, Jingumae 5-chome
Shibuya-ku, Tokyo 150-8925
uchida@undl.org

Meiying Zhu
UNL Center
UNDL Foundation
53-70, Jingumae 5-chome
Shibuya-ku, Tokyo 150-8925
zhu@undl.org

Introduction

When computers make intelligent processing based on knowledge including reasoning, enough knowledge must be provided in the form that accessible and understandable for computers. We call the collection of such knowledge “Knowledge Infrastructure (KI)”. In this article, we describe how UNL (Universal Networking Language) represent knowledge written in natural languages to provide KI, and how such knowledge to be utilized for intelligent processing by computers.

UNL is a language for computer to express information and knowledge that can be expressed in natural languages. The UNL has been developed to express any kind of knowledge that can be represented in any natural languages and to make it possible that such knowledge can be fully accessible in any natural languages. For this purpose UW (Universal Word) System (ontology) has been developed allowing people to introduce and define any concept (UW) no matter how particular or specific to a language it is. All UWs defined in the UW System can be understood by computers by describing all possible relations that every UW has with other UWs in UNLKB. Such UNLKB not only provides linguistic knowledge in the form that computer can understand but also provides the semantic basis of UNL expressions of sentences.

For the purpose that UNL can replicate, in the cyber world, the functions of natural languages in human communication, the UNL should provide the mechanism to infer “John is the author of the book” from “John wrote the book” for example. In this inference, not only the knowledge of meaning of the words used in the sentences, but also the knowledge about the concept provided by the sentences such as “An author is a person who write a book” is necessary. Various kinds of knowledge are provided in natural languages, and those kinds of knowledge need to be provided for computers..

In this article we first describe how we view the various knowledge represented in natural languages and how we

organize the knowledge in UNL. And then we describe what kinds of knowledge are necessary and how they should be represented in UNL for reasoning by computers.

This article consists of the following sections:

- Introduction
- Formalization of Knowledge Representation in UNL
- Outline of UNL and UNL System
- Logic Expression in UNL
- Conclusion

Categories and Subject descriptions:

I.2.4 Knowledge Representation Formalisms and Methods (e.g. Representation languages, Semantic networks)
I.2.7 Natural Language Processing (e.g. Language models)

General Terms: Languages

Keywords: Meaning Representation, Knowledge Representation, Knowledge Structure, Logical Expression, UNL

1. Formalization of Knowledge Representation in UNL

We consider that knowledge provided in natural languages can be divided into three levels: linguistic knowledge, definitions of concepts, and others. Linguistic knowledge is the knowledge providing semantic information of words for understanding sentences of natural languages. For example, an “author” is a “person”, which can take various actions that a person can take. This level of knowledge is necessary to provide semantic background of natural language sentences. Definitions of concepts provide the knowledge of the concepts in connection with other concepts that can specify the concepts. For example “a person who write a book” is the sentence to define what is an “author”. Through this definition, it is possible to know that a person who wrote a book will be the author of the book. This level of knowledge is indispensable for reasoning in information retrieval. etc.. The

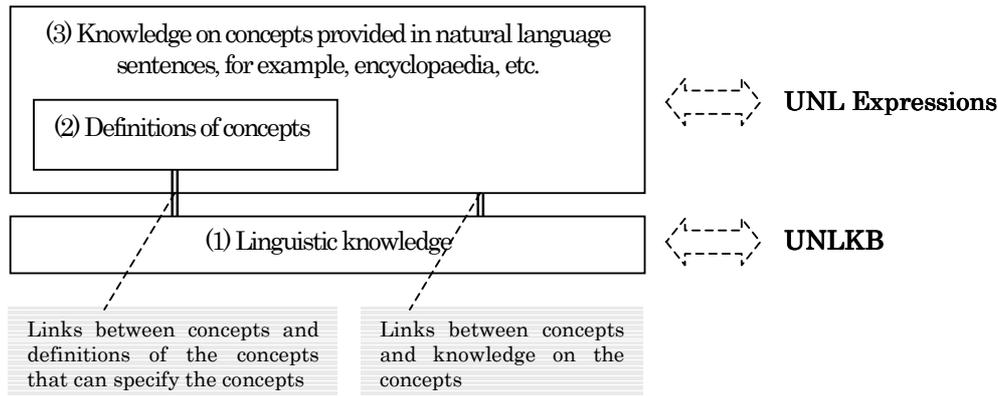


Figure 1. Three levels of knowledge in the UNL Knowledge System

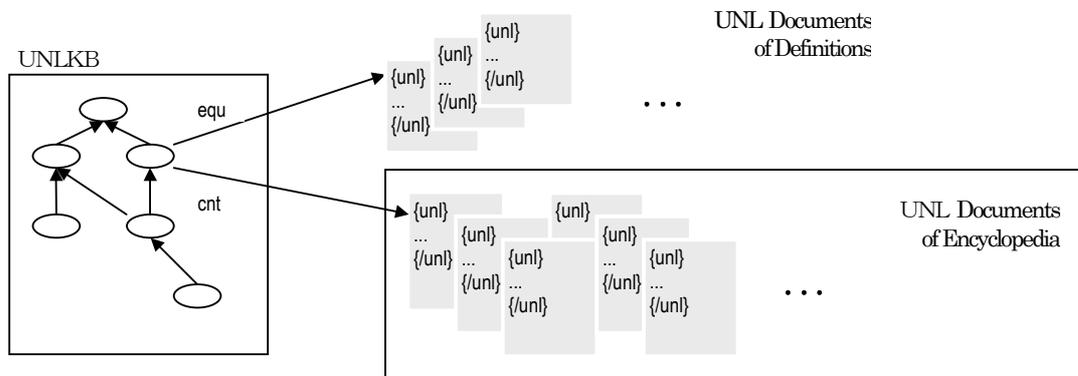


Figure 2. UNL Expressions and relations of three levels of knowledge in the UNL Knowledge System

third level of knowledge includes every kind of knowledge provided by natural language sentences such as encyclopaedia. This level of knowledge is for providing human with necessary information of concepts, and for providing computers with various information and knowledge on concepts necessary for reasoning and providing information on the concepts. In the UNL system, linguistic knowledge is provided in UNLKB with UWs representing concepts. Definitions of concepts and knowledge on concepts described in natural language sentences are represented by UNL expressions. Figure 1 shows the relations of the three levels of knowledge and how they are represented in the UNL Knowledge System.

Figure 2 shows how the three levels of knowledge are represented in UNL and how they are linked with each other in the UNL Knowledge System.

- UWs representing concepts of words are defined in UNLKB. The circles of nodes indicate UWs and arrows indicate relations in figure 2. Concepts of UWs are defined in the UW System of the UNLKB. The UW System is a lattice type of hierarchy made of UWs related to each other. The details of the UW System and

UNLKB are explained in section 2.1.

- A definition of a concept (UW) is described in a file. Each file consists of sentences to define the concept of the UW. Sentences are expressed in UNL. Each definition file is linked with corresponding UW using relation 'equ'. All UWs used in the UNL expressions of definition must be defined in UNLKB.
- UNL encyclopaedia is composed of files of sentences (UNL documents) describing information and knowledge about UWs. A file contains information and knowledge about a UW. Information and knowledge about a UW can be provided in different files. Each file of encyclopaedia is differentiated using the developer name of the content as the file name. These files are linked with corresponding UWs using relation 'cnt'. All UWs included in UNL expressions of information and knowledge must be defined in UNLKB.

UWs are linked with definition files or encyclopaedia files in the following format:

```
equ ( <UW>, <UNL Document of Definition> )
cnt ( <UW>, <UNL Document of Encyclopaedia> )
```

Where,

<UW> ::= a UW defined in the UNLKB

<UNL Document of Definition> ::= a local file or a URL of a UNL document

<UNL Document of Encyclopedia> ::= a local file or a URL of a UNL document

A UNL document means a text file or a HTML file in which UNL expressions for the original natural sentences are included. A UNL document is attached to with extension “.unl” as in “http://www.undl.org/UNLencyclopedia/GreatWall.html.unl” or “x:/unlencyclopedia/local/GreatWall.html.unl”.

2. Outline of the UNL and the UNL System

This section explains the UNL and its operating system: UNL System.

2.1 UNL: Universal Networking Language

UNL expresses information or knowledge in the form of semantic network with hyper-node. Different from natural languages, UNL expressions are unambiguous. In the UNL semantic network, nodes represent concepts, and arcs represent relations between concepts. Concepts can be annotated.

Since the UNL is a language for computers, it has all the components of a natural language. It is composed of words expressing concepts called “Universal Words”, also referred to as UWs that are inter-linked with other UWs to form sentences. These links, known as “relations”, specify role of each word in a sentence. The subjective meaning intended by the speaker can be expressed through “attributes”.

The “Knowledge Base (UNLKB)” is provided to define semantics of UWs. The UNLKB defines every possible relation between concepts including hierarchical relations and inference mechanism based on inclusion relations between concepts. Thus, the UNLKB provides semantic background of the UNL to make sure the meaning of the UNL expressions is unambiguous.

Format of UNL Expression

The UNL expresses information or knowledge in the form of semantic network with hyper-node. UNL semantic network is made up of a set of **binary relations**, each binary relation is composed of a relation and two UWs that hold the relation. A binary relation of UNL is expressed in the following format:

<relation> (<uw1>, <uw2>)

In <relation>, one of the relations defined in the UNL Specifications is described. In <uw1> and <uw2>, the two UWs that have the relation given by <relation> are described. Semantic network of UNL expression is a **directed graph** by means of the binary relations with direction. The three elements of each binary relation have the following interrelationship:

<uw1> – <relation> → <uw2>

Such a binary relation is **interpreted** as that: the UW given in <uw2> plays the role indicated by the relation given in <relation> held by the UW given in <uw1>; whereas the UW given in <uw1> holds the relation given in <relation> with the UW given in <uw2>.

A UNL expression is a **hyper semantic network**. That is, each node of the graph, <uw1> and <uw2> of a binary relation, can be replaced with a semantic network. Such a node consists of a semantic network of a UNL expression and is called a “scope”. A scope can be connected with other UWs or scopes. The UNL expressions in a scope is distinguished from others by assigning an ID to the <relations> of the set of binary relations that belong to the scope.

The **general description format** of binary relations for a hyper-node of UNL expression is the following:

<relation> : <scope-id> (<node1>, <node2>)

Where,

<scope-id> is the ID for distinguishing a scope.

<scope-id> is not necessary to specify when a binary relation does not belong to any scope.

<node1> and <node2> can be a UW or a <scope node>.

A <scope node> is given in the format of “<scope-id>”.

Relations

There are many factors to be considered in choosing an inventory of relations between concepts. Different factors taken into account in choosing the relations lead to different sets of the relations. The UNL relations (listed in Table1) are selected basically according to the following principles:

PRINCIPLE 1 : NECESSARY CONDITION

When an UW has relations between more than one other UWs, each relation label should be set so as to be able to identify each relation on the premise that there is enough knowledge about the concept of each UW expressed.

PRINCIPLE 2 : SUFFICIENT CONDITION

When there are relations between UWs, each relation label

should be set so as to be able to understand the role of each UW only by referring to the relation label.

Universal Words : UWs

Universal Words are words that constitute the vocabulary of UNL. A Universal Word is not only a unit of the UNL syntactically and semantically for expressing a concept, but also a basic element for constructing a UNL expression of a sentence or a compound concept. Such a Universal Word is represented as a node in the hyper-graph of UNL expressions.

Syntax of UW

A UW is made up of a character string (an English word) followed by a list of constraints. The meaning and function of each of these parts is described in the next section.

The following is the syntax of description of UWs:

```
<UW> ::= <headword> [<constraint list>]
<headword> ::= <character>...
<constraint list> ::= (“<constraint> [ “,” <constraint>]... ”)
<constraint> ::= <relation label> { “>” | “<” } <UW> [<constraint list>] |
    <relation label> { “>” | “<” } <UW> [<constraint list>]
    [ { “>” | “<” } <UW> [<constraint list>] ] ...
<relation label> ::= “agt” | “and” | “aoj” | “obj” | “icl” | ...
```

The **headword** is an English word/compound word/phrase/sentence that is interpreted as a label for a set of concepts: the set made up of all the concepts that may correspond to that in English. A Basic UW (with no restrictions or constraint list) denotes this set. Each Restricted UW denotes a subset of this set that is defined by its constraint list. Extra UWs denote new sets of concepts that do not have English labels. Thus, the headword serves to organize concepts and make it easier to remember which is which.

The **constraint list** restricts the interpretation of a UW to a subset or to a specific concept included within the Basic UW, thus the term “Restricted UWs”. The Basic UW “drink”, without a constraint list, includes the concepts of “putting liquids in the mouth”, “liquids that are put in the mouth”, “liquids with alcohol”, “absorb” and others. The Restricted UW ‘drink(agt>thing,obj>liquid)’ denotes the subset of these concepts that includes “putting liquids in the mouth”, which in turn corresponds to verbs such as “drink”, “gulp”, “chug” and “slurp” in English.

A restriction of a UW is made up of a pair of a relation and a defined UW (or part expression of a defined UW) that holds the relation with this UW. If more than one restrictions are necessary, a comma “,” should be used between restrictions.

A Restricted UW is defined through a Master Definition¹. In a Master Definition, full expressions of defined UWs must be described in the restrictions, whereas as for a UW, if and only if the uniqueness can be kept, part of the pre-defined UWs (its headword or part restrictions) can be used in the restrictions. Relation labels used in the constraint list must be defined in the UNL specifications and should be sorted in alphabetical order if more than one restriction are used.

In order to define the meaning of a UW more accurately, for instance, a subset concept of UW is always defined under an upper UW that has the closest but more general meaning. This is implemented by linking the UW to be defined with the upper UW using ‘icl’ relation. For example, UW ‘provide(icl>give(agt>thing,gol>thing,obj>thing))’ is defined as a subset concept of UW ‘give(agt>thing,gol>thing,obj>thing)’. However, if the headword of the upper UW is either of “be”, “do”, “occur” and “uw”, such a headword is not necessary to remain in the restrictions of lower UWs as the each set of restrictions of these upper UWs is set enough to restrict their lower UWs. For example, from Master Definition ‘drink({icl>do({agt>thing,obj>liquid})})’ a UW ‘drink(agt>thing,obj>liquid)’ and a binary relation ‘icl(drink(agt>thing,obj>liquid), do(agt>thing,obj>liquid))’ are generated. The part related to the headword “do” is removed from its lower UW expression and the binary relation that will be described in the UNLKB shows that ‘drink(agt>thing,obj>liquid)’ is a subset concept of ‘do(agt>thing,obj>liquid)’. For details of description of UW please refer to UW manual.

Types of UW

A UW is a character string and most of the UWs are basically made up of an English expression with restrictions. A UW can express various levels’ concepts depending on the restrictions and can be used to express a more specific or particular concept or an instance by giving attributes and IDs or restrictions from other UNL expressions. The UWs are divided into four types:

- Basic UWs, which are bare headwords with no constraint list, for example:
 - go
 - take
 - house
 - state
- Restricted UWs, which are headwords with a constraint list, for example:
 - state(icl>express(agt>thing,gol>person,obj>thing))
 - state(icl>country)

¹ For details please refer to UW Manual at <http://www.unl.org/unlsys/>

state(icl>region)
state(icl>abstract thing)
state(icl>government)

- Extra UWs, which are a special type of Restricted UW, for example:
ikebana(icl>flower arrangement)
samba(icl>dance)
soufflé(icl>food)
- Temporary UWs, which are not necessary to define, for example:
1234
xyz

Attributes

Attributes are mainly for the purpose to describe the subjectivity information of sentences. They show what is said from the speaker's point of view: how the speaker views what is said. This includes phenomena technically called "speech acts", "propositional attitudes", "truth values", etc. Attributes are also used to express the range of concepts such as the concept indicate generic type of concept and so forth. This time, we newly introduce attributes to express logical expressions in order to strengthen the expressibility of the UNL.

Relations and UWs are used to describe the objectivity information of sentences. Attributes modify UWs or semantic networks (scope: compound concepts) to indicate subjectivity information such as about how the speaker views these states-of-affairs and his attitudes toward them and to indicate the property of the concepts.

Attributes are divided into the following groups:

- Describing logic characters / properties of concepts
- Times with respect to the speaker
- Speaker's view on aspects of event
- Speaker's view of reference to concepts
- Speaker's view of emphasis, focus and topic
- Speaker's attitudes
- Speaker's feelings and judgments

Attributes of UNL are listed in tables 2 ~ 6 and 8. Tables 6 and 8 are shown in section 3.2.

UNLKB

The UNLKB is a semantic network comprising every directed binary relation between UWs. All the binary relations of the UNL KB are in the following format: 'relation(UW1, UW2)=c', where 'c' is the degree of certainty, which has the value 0 (impossible) or 1 to 128 (certain). This binary relation means "UW1 takes UW2 as the relation in certainty value c", or "UW2 plays the role specified by relation for UW1 in certainty

value c".

- The UNLKB Defines Concepts of UWs

The UNLKB provides concepts and UW of the concepts. A UW (Universal Word) is a label for a concept. Concepts labelled by UWs are defined by describing the set of possible relations that each concept can have with other concepts. Definitions of possible relations with other concepts describe behaviours of concepts in terms of other concepts. This behaviour is the property of a concept in the sense that the descriptions of behaviour characterize the concept and provide enough information to understand the semantic structure of the sentence.

- The UNLKB Provides Linguistic Knowledge of Concepts

The behaviour of a concept is considered as linguistic knowledge on the concept. This knowledge is used to provide semantic structure of sentences of natural languages. For example, an "author" is a "person", which can take various actions that a person can take such as a person can writes something and something can be a book, and so forth. This level of knowledge is necessary to provide semantic background of natural language sentences. Further knowledge, for example real world knowledge and so forth, will be established based on this linguistic knowledge using the UWs.

UW System

In the UNL KB, all UWs are linked with each other through 'icl' (subclass) 'iof' (element/instance), or 'equ' relations. 'icl' links a UW of a subclass concept to a super-class concept UW; 'iof' links a UW expressing an instance to a UW of a class concept; The UWs related to each other through 'icl', 'iof' and 'equ' relations make up a hierarchy of UWs. This hierarchy of UWs is the UW system. This UW system allows having multiple super-class concepts. Accordingly, the UW system is a lattice type of network.

The hierarchy of the UW system is constructed by taking the property inheritance and replacement by super-class concept mechanisms into consideration. In UW system, lower UWs inherit the properties of upper UWs; and upper UWs can replace lower UWs to convey a more general sense in the specific context of the lower UWs. All these inheritance and replacement are carried out through the relations 'icl', 'iof' and 'equ'.

In the UNL KB, all possible relations, such as 'agt', 'obj', etc, that an UW can have with others are defined for each. Every possible relation is defined between the two most general UWs of the two categories (of lower UWs) that can have the relation. Utilizing the property inheritance mechanism of the

UW system possible relations of lower concepts are deductively inferred and this inference mechanism can reduce the number of binary relations.

Replacement of lower UWs by upper UWs can cause problems by introducing ambiguities if the upper UWs are not close in meaning to the lower UWs. To avoid this, the upper UWs must be the closest UWs among all of the more general UWs. In other words, every UW must be positioned under the closest upper UWs.

The UNLKB defines the syntax and semantics of the language UNL. In the UNL System, UNLKB is used in sentence analysis for disambiguation and in sentence generation for finding more general concepts when encountering unknown concepts to the language. The UNLKB also is used to verify UNL expressions since it provides syntax and semantics of UNL.

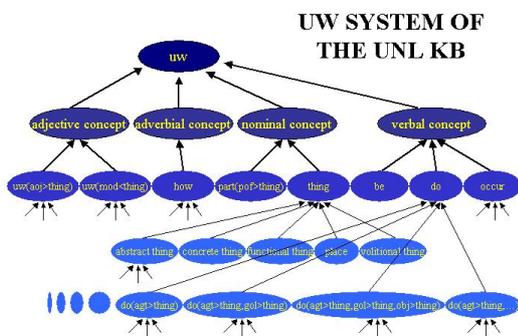


Figure 3

Table 1: relations of UNL

agt	agent	a thing in focus that initiates an action
and	conjunction	a partner to have conjunctive relation
aoj	thing with attribute	a thing that is in a state or has an attribute
bas	basis	a thing used as the basis (standard) of comparison
ben	beneficiary	an indirectly related beneficiary or victim of an event or state
cag	co-agent	a thing not in focus that initiates an implicit event that is done in parallel
cao	co-thing with attribute	a thing not in focus that is in a parallel state
cnt	content	the content of a concept
cob	effected co-thing	a thing that is directly affected by an implicit event done in parallel or an implicit state in parallel
con	condition	a non-focused event or state that conditions a focused event or state
coo	co-occurrence	a co-occurrent event or state for a focused event or state
dur	duration	a period of time during which an event occurs or a state exists
equ	equivalent	an equivalent concept
fmt	range	a range between two things
frm	origin	an initial state of a thing or a thing initially associated with the focused thing
gol	goal/final state	a final state of object or a thing finally associated

icl	included/ a kind of	with the object of an event an upper concept or a more general concept
ins	instrument	a instrument to carry out an event
int	intersection	a partner to take an intersection
iof	an instance of	a class concept that an instance belongs to
man	manner	a way to carry out an event or characteristics of a state
met	method	means to carry out an event
mod	modification	a thing that restricts a focused thing
nam	name	a name of a thing
obj	effected thing	a thing in focus that is directly effected by an event or state
opl	effected place	a place in focus affected by an event
or	disjunction	a partner to have disjunctive relation
per	proportion, rate of distribution	a basis or unit of proportion, rate of distribution
plc	place	the place where an event occurs, or a state is true, or a thing exists
plf	initial place	the place where an event begins or a state becomes true
plt	final place	the place where an event ends or a state becomes false
pof	part-of	a concept of which a focused thing is a part
pos	possessor	a possessor of a thing
ptn	partner	an indispensable non-focused initiator of an action
pur	purpose or objective	the purpose or objective of an agent of an event or the purpose of a thing which exists
qua	quantity	quantity of a thing or unit
rsn	reason	a reason why an event or a state happens
scn	scene	a scene where an event occurs, or state is true, or a thing exists
seq	sequence	a prior event or state of a focused event or state
src	source/initial state	the initial state of an object or thing initially associated with the object of an event
tim	time	the time an event occurs or a state is true
tmf	initial time	the time an event starts or a state becomes true
tmt	final time	the time an event ends or a state becomes false
to	destination	a final state of a thing or a final thing (destination) associated with the focused thing
via	intermediate place or state	an intermediate place or state of an event

Table 2: attributes expressing times and aspects

@past	happened in the past
@present	happening at present
@future	will happen in future
@begin	beginning of an event or a state
@complete	finishing/completion of a (whole) event.
@continue	continuation of an event
@custom	customary or repetitious action
@end	end/termination of an event or a state
@experience	experience
@progress	an event is in progress
@repeat	repetition of an event
@state	final state or the existence of the object on which an action has been taken

Table 3: attributes expressing emphasis, focus, etc

@contrast	Contrasted UW
@emphasis	Emphasized UW
@entry	Entry or main UW of a sentence or a scope
@qfocus	Focused UW of a question

@theme	Instantiates an object from a different class
@title	Title
@topic	Topic

Table 4: attributes expressing speaker's attitudes

@affirmative	Affirmation
@confirmation	Confirmation
@exclamation	Feeling of exclamation
@humility	In a humility manner to express something
@imperative	Imperative
@interrogative	Interrogation
@invitation	Inducement to do something
@polite	Polite way to express something
@request	Request
@respect	Respectful feeling
@vocative	Vocative

Table 5: attributes expressing speaker's feelings and judgments

@ability	Ability, capability of doing something
@admire	Admiring feeling of the speaker about something
@although	Something follows against [contrary to] or beyond expectation
@blame	Blameful feeling of the speaker about something
@certain	Certainty that something is true or happens
@conclusion	Logical conclusion due to a certain condition
@contempt	Contemptuous feeling of the speaker about something
@consent	Consent feeling of the speaker about something
@consequence	Logical consequence
@discontented	Discontented feeling of the speaker about something
@disjointed	connected concept do not share instances
@dissent	Dissent feeling of the speaker about something
@expectation	Expectation of something
@get-benefit	Speaker's feeling of receiving benefits through the fact or result of something (to be) done by somebody else
@give-benefit	Speaker's feeling of giving benefits by doing something for somebody else
@grant	To give/get consent/permission to do something
@grant-not	Not to give consent to do something
@identifiable	identify the subject
@inevitable	Logical inevitability that something is true or happens
@insistence	Strong determination to do something
@intention	Intention about something or to do something
@may	Practical possibility that something is true or happens
@need	Necessity to do something
@obligation	Obligation to do something according to (quasi-) law, contract, or ...
@obligation-not	Obligation not to do something, forbid to do something according to (quasi-) law, contract or ...
@possible	Logical possibility that something is true or happens
@probable	(Practical) probability that something is true or happens
@rare	Rare logical possibility that something is true or happens
@regret	Regretful feeling of the speaker about something
@should	To do something as a matter of course
@sufficient	Sufficient condition
@surprised	Surprised feeling of the speaker about something
@symmetric	has symmetricity
@transitive	has transitivity
@troublesome	Troublesome feeling of the speaker about the occurrence of something

@unavoidable	Unavoidable feeling of the speaker about doing something
@unreal	Unreality that something is true or happens
@want	Desire to do something
@will	Determination to do something
@wish	Wishful feeling, to wish something is true or has happened

2.2 The UNL System

The UNL System consists of three major components: language resources, software for processing the language resources, and tools and systems for maintaining and operating the language processing software or language resources. Language resources are divided into language dependent part and language independent part. Linguistic knowledge on concepts that universal to every language is considered language independent and to be stored in the common database UNLKB. Language dependent resources like word dictionaries and rules, as well as the software for language processing, are stored in each language server. Language servers are connected in the Internet through UNL. Supporting tools for producing UNL documents can be used in a local PC. Such supporting tools function with consulting language servers through the Internet. Verification of UNL documents can be carried out through the Internet or in a local PC. The tool (UW Gate) for search or maintenance of the common database UNLKB functions through the Internet.

The structure of the UNL System

In figure 4, the highlighted parts show the components of the UNL System. White parts show applications and their UNL database based on the UNL System.

Each component of figure 4 is explained below, from the upper right:

UNLKCIC stores information of Key Concept in Context (KCIC) about UNL documents. The KCIC is made for every binary relation of UNL documents. This information is used when searching for related UNL expressions of a UNL expression. Through UNLKCC, each UWs in the UNLKB is linked to the instance of it, then world knowledge on UNL is described as relations between other UWs.

Concept Definitions are the collection of definitions of UWs. Definitions are provided by sentences and are expressed in UNL (document). Definitions (semantics/meaning) of concepts provide the knowledge of the concepts in connection with other concepts that can specify the concepts. This knowledge is indispensable for reasoning in information retrieval, etc.

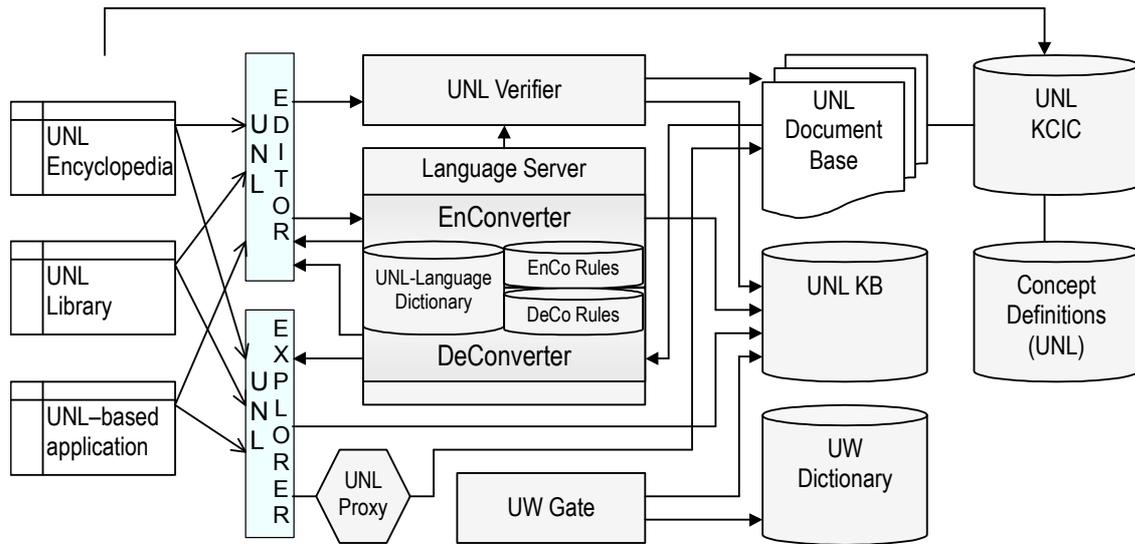


Figure 4. Structure of the UNL System

UNL Documents mean the documents in which UNL expression is described for each sentence of natural language. A UNL document can be made of a plain text file or an UNL-embedded html file. A UNL Document base is a collection of UNL Document files. UNL documents are for the purpose to provide information in UNL.

The **UNLKB** is explained in previous section 2.1.

UW Dictionary provides the interface between UWs and words of natural languages.

The **UNL Verifier** verifies whether a UNL expression is correct syntactically, lexically and semantically. The syntax check of a UNL expression is carried out against the UNL Specifications. In lexical check, whether all UWs of a UNL expression are defined in the UNLKB are checked. In semantic check, whether each binary relation of a UNL expression is defined as possible is certified with consulting the UNLKB.

UNL Language Servers (LSs) are located in the Internet to carry out the conversions between natural languages and UNL expressions. Each LS contains an EnConverter and a DeConverter of a language. EnConverter converts natural language sentences to UNL expressions. DeConverter convert UNL expressions to natural language sentences.

The **UW Gate** is tool for people to access the UNLKB and the UW dictionary through the Internet. The use of the UW gate is authorized to the UNL Society members in three levels. First, every member of the UNL Society is allowed to search the UNLKB and UW dictionary for necessary information. Second, members that belong to a particular language group

are allowed to modify the links between the language and UWs. Third, members that have mastered how to define UWs are allowed to register new UWs to the UNLKB.

The **UNL Proxy Server** functions to communicate with language servers. It functions as a filter to check whether a web page that a user required is written in UNL or not. If UNL expressions are included in the web page, it communicates with an appropriate language server in the Internet for deconverting the UNL expressions into desired language sentences and provides the Internet browser with the results to display.

The **UNL Editor** is a tool helps to produce UNL documents. It includes an EnConverter and a DeConverter. Each of them can be selected according to language. EnConverter converts natural language sentence into UNL expressions. Whereas DeConverter provides generated results as feedback for checking the correctness of UNL expressions.

The **UNL Explorer** provides the basis of knowledge infrastructure. The UNL Explorer manages UNLKB, UNLKCIC and UNL Documents provide the knowledge on UWs. The UNL Explorer has two types of function. For human, the UNL Explorer allows users or developers to view or to develop the UNL Knowledge System such as the UNL Encyclopaedia. For computers, it provides information or knowledge on UWs.

The UNL Explorer uses UNLKB for navigating information stored in UNL database. It has two windows: the hierarchy of UWs (UW System) of UNLKB is shown in the left window. UWs of the UNLKB are keys for information stored in UNL database. Information on UWs is shown in the right window

through navigation through the UW System. Information on UWs is described in UNL documents. All UWs used in the UNL documents are included in the left window of UNLKB and are keys for further information.

The UNL Explorer allows users to search for information using UWs or words of natural languages. It shows the information in UNL or a desired natural language by accessing UNL Language Servers. It also provides functions for developers to add information to or modify information of the UNL database in their native languages.

Information about a UW is stored in a file. Location of the file is linked with the UW. This architecture of the UNL Knowledge System allows its development to be carried out by a wide range of developers from different languages and cultures. Such a database can provide a wealth of up-to-date information on various aspects of information and knowledge from all over the world.

UNL Library is an UNL-based application providing bibliographic information (e.g. MARC21) and contents in future in UNL, and allowing users to search for books through UNL.

Mechanism of conversion of UNL expressions

Figure 5 shows the mechanism how a UNL document is

made and how a UNL document is converted into natural languages in the UNL system.

The EnConverter and DeConverter are the core software in the UNL system. The EnConverter converts natural language sentences into UNL expressions. The Universal Parser (UP) is a specialized version of the EnConverter. It generates UNL expressions from annotated sentences using the UW dictionary only. All UNL expressions are verified by the UNL verifier. The DeConverter converts UNL expressions to natural language sentences.

Figure 6 shows an example of use of the UNL System.

3. Logic Expression in UNL

In this section we describe how UNL express logic and what expressive functions the UNL should have for logic expressions and reasoning. Reasoning using the UNL and the expressions of UNL for logic has not been mentioned much before. Recently, many applications based on the UNL are being developed. To clarify the logic expression in UNL and mechanism of reasoning using UNL become important for the UNL applications especially for knowledge based systems.

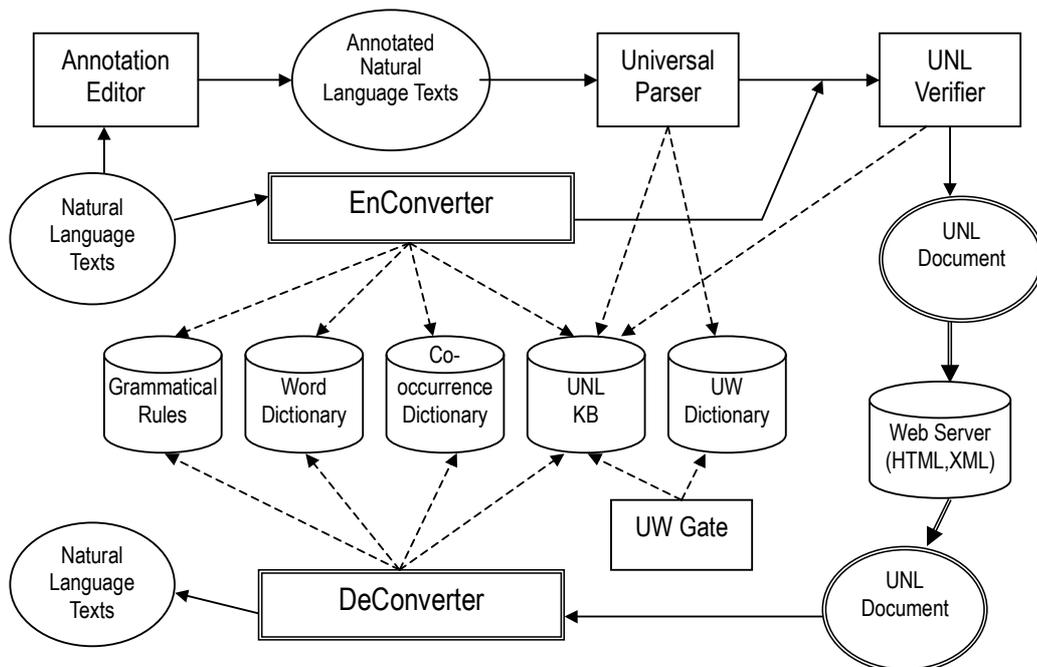


Figure 5. Mechanism of conversion of UNL expressions

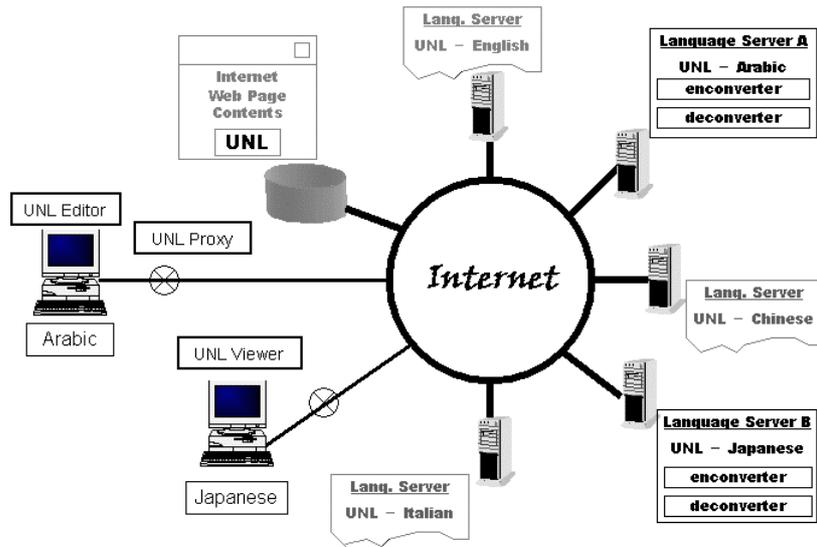


Figure 6. An example of use of the UNL System

3.1 How Variables are expressed in UNL

Every concept is represented using UW in UNL. For example, inference rules or formulas for information retrieval are necessary to be expressed in UNL also. Variables are often used in a formula or rule. UNL has to deal with variables. Variables are expressed as temporary UWs with variable names as headwords. The domains of variables are expressed as constraints in UWs, such as 'x(iof>domain)'. In case of "x ∈ dog", a variable in UNL will be 'x(iof>dog)'.

[ex1] Logical expression for "dogs are clever" using variable in UNL will be the following:

$$\forall x (\text{dog}(x(\text{iof}>\text{dog})) \rightarrow \text{clever}(x(\text{iof}>\text{dog}))) \quad (1)$$

The expressions of (1) is equivalent to (1)':

$$\forall x \in \text{dog} (\text{dog}(x) \rightarrow \text{clever}(x)) \quad (1')$$

However, for expressing the logical expression like "∀ x,y (x=parent(y) → y=child(x))", more than one variables are necessary to introduce.

[ex2] Logical expression for "∀ x,y (x=parent(y) → y=child(x))" using variables in UNL is the following:

$$\forall x,y (x(\text{iof}>\text{parent})=\text{parent}(x(\text{iof}>\text{child})) \rightarrow y(\text{iof}>\text{child})=\text{child}(x(\text{iof}>\text{parent}))) \quad (2)$$

UNL expressions of the logical expressions (1) and (2) are the following:

```
[UNL1]
aoj:01 ( dog(icl>mammal).@entry, x(iof>dog) )
con ( :02.@entry, :01 )
```

```
aoj:02 ( clever(aoj>thing).@entry, x(iof>dog) )
```

```
[UNL2]
aoj:01 ( parent(icl>kinfolk).@entry, x(iof>parent) )
pos:01 ( parent(icl>kinfolk).@entry, y(iof>child) )
con ( :02.@entry, :01 )
aoj:02 ( child(icl>kinfolk).@entry, y(iof>child) )
pos:02 ( child(icl>kinfolk).@entry, x(iof>parent) )
```

Relation 'con' has the same function as the logical operator → (implication).

UNL Expression for Logical Expression and Sentence

What [UNL1] shows directly is "if x is a dog, it is clever". However it is the UNL expression of the logical expression of "dogs are clever". This means that "if x is a dog, it is clever" is equivalent to "dogs are clever", and [UNL1] is equivalent to the [UNL1-1] below.

```
[UNL1-1]
aoj ( clever(aoj>thing).@entry, dog(icl>mammal) )
qua ( dog(icl>mammal), all(mod<thing) )
```

From this, we can say that [UNL1] is more appropriate for reasoning; whereas [UNL1-1] is more simple and more appropriate for natural language analysis and generation. Generally, knowledge provided by natural languages is expressed by UNL in the simple way like [UNL1-1]. However, sometimes the logical expressions need to be extracted from knowledge representation of UNL for conducting reasoning. Likewise, logical expressions of results of reasoning should be converted into simple UNL expressions so that more natural sentences can be generated easily.

For instance, conversion between [UNL1] and [UNL1-1] can be carried out in the following steps:

Steps of conversion from [UNL1] to [UNL1-1]:

- $X(\text{iof}>\text{dog})$ is an variable in the condition part $\text{aoj}(\text{dog}(\text{icl}>\text{mammal}), x(\text{iof}>\text{dog}))$ and the relation is "aoj", we can get $x(\text{iof}>\text{dog}) = \text{dog}(\text{icl}>\text{mammal})$.
- In addition to this, $x(\text{iof}>\text{dog})$ is a variable and can have all the element of $\text{dog}(\text{icl}>\text{mammal})$, $\text{qua}(\text{dog}(\text{icl}>\text{mammal}), \text{all}(\text{mod}<\text{thing}))$ is added. Then substitutes the variable $x(\text{iof}>\text{dog})$ for the value of the variable $\text{dog}(\text{icl}>\text{mammal})$, we can get $\text{aoj}(\text{clever}(\text{aoj}>\text{thing}).@entry, \text{dog}(\text{icl}>\text{mammal}))$.

Steps of conversion from [UNL1-1] to [UNL1]:

- Introduce a variable $x(\text{iof}>\text{dog})$ to represent all the element of $\text{dog}(\text{icl}>\text{mammal})$ based on the binary relation $\text{qua}(\text{dog}(\text{icl}>\text{mammal}), \text{all}(\text{mod}<\text{thing}))$ and make a condition part $\text{aoj}:01(\text{dog}(\text{icl}>\text{mammal}).@entry, x(\text{iof}>\text{dog}))$ and also make conclusion part $\text{aoj}:02(\text{clever}(\text{aoj}>\text{thing}).@entry, x(\text{iof}>\text{dog}))$, then generate linking $\text{con}(:02.@entry, :01)$.

Generalization of conversion rules between knowledge representation in UNL and logical expression in UNL will be one of the research themes of the future.

3.2 UNL expression on quantification of logical expression

Concepts of UWs can be divided into class concepts and instance. UWs for expressing class concepts can be used to express all instances of the class, a part of the class or a particular instance. In UNL, quantities of concepts are expressed by UWs or attributes that have the function to indicate the quantity.

Table 6: Attributes that have the functions for quantification

@generic	Attached to a class concept of UW to express all elements of the concept
@def	Attached to a class concept of UW to express some particular instance(s) of the concept. -> Existence
@indef	Attached to a class concept of UW to express existence of some instance(s) of the concept. ->Existence.
@not	Attached to a class concept or an instance of UW to express the complement concepts or instances. -> Existence
@ordinal	Attached to a class concept of UW to express some particular instance(s) of the concept. -> Existence

Table 7: Example of UWs expressing a quantity

every(mod<thing)	Has 'mod' relation with a class concept UW to express all instances of the concept
some(mod<thing)	Has 'mod' relation with a class concept UW to express existence of some instances of the concept
no(mod<thing)	Has 'mod' relation with a class or instance concept UW to express no instance of the concept exists
most(mod<thing)	Has 'mod' relation with a class concept UW to express most of instances of the concept
many(qua<thing)	Has 'qua' relation with a class concept UW to express existence of many instances of the concept
<a number>	Have 'qua' relation with a class concept UW to express the number of instances of the concept

Note: <a number> means a numeral number like 205, 2, etc

The followings are examples to show how the universal quantifier(\forall) and existential quantifier(\exists) of logical expressions are expressed in UNL.

[ex3] Logical expression for "all is walking" using variables in UNL is the following:

$$\forall x(\text{iof}>\text{person}) (\text{person}(x(\text{iof}>\text{person})) \rightarrow \text{walk}(x(\text{iof}>\text{person}))) \quad (3)$$

[UNL3] shows the UNL logical expression (3), and [UNL3-1] shows the UNL semantic expression of the sentence.

[UNL3]
 $\text{aoj}:01(\text{person}(\text{icl}>\text{human}).@entry, x(\text{iof}>\text{person}))$
con (:02.@entry, :01)
 $\text{agt}:01(\text{walk}(\text{agt}>\text{thing}).@entry, x(\text{iof}>\text{person}))$

[UNL3-1]
 $\text{agt}(\text{walk}(\text{agt}>\text{thing}).@entry, \text{person}(\text{icl}>\text{human}))$
 $\text{qua}(\text{person}(\text{icl}>\text{human}), \text{all}(\text{mod}<\text{thing}))$

[ex4] Logical expression for "someone is walking" using variables in UNL is the following:

$$\exists x(\text{iof}>\text{person}) (\text{person}(x(\text{iof}>\text{person})) \wedge \text{walk}(x(\text{iof}>\text{person}))) \quad (4)$$

UNL logical expression:

[UNL4]
 $\text{aoj}:01(\text{person}(\text{icl}>\text{human}).@entry, x(\text{iof}>\text{person}))$
int (:02.@entry, :01)
 $\text{agt}:01(\text{walk}(\text{agt}>\text{thing}).@entry, x(\text{iof}>\text{person}))$
 Relation 'int' of UNL has the same function as intersection (\wedge) of logical operator.

UNL semantic expression of the sentence:

[UNL4-1]
 $\text{agt}(\text{walk}(\text{agt}>\text{thing}).@entry, \text{person}(\text{icl}>\text{human}).@indef)$

3.3 Attributes and relations for logical expression and reasoning

Attributes of UNL have been developed mainly to express subjectivities information of the speaker of a sentence. In UNL2005, we introduced new attributes to express logical expressions in order to strengthen the expressibility of the UNL although the role of those attribute can be expressed by using UWs as ordinal UNL expressions. In this section, we focus on describing attributes and relations for expressing logic.

Attributes for expressing logicity of UW

Table 8 shows the attributes for expressing logicity of UW. These attributes are used to attach to the UWs that have the logicity. '@disjointed' can be attached to a group of UWs (a scope) to show that every concept included in the group are disjointed. It means that all element concepts do not share instances.

Table 8: attributes for expressing logicity of UW

@transitive	attached to an UW that has transitivity
@symmetric	attached to an UW that has symmetry
@identifiable	attached to an UW that can identify the subject
@disjointed	attached to an UW or a group of UWs to show that all element concept do not hold common instance. All connected UWs do not share instances.

- '@transitive' can be attached to UW 'ancestor(ic>kindred)'. Because if "A is ancestor of B" and "B is ancestor of C" are true, "A is ancestor of C" will be true.
- '@symmetric' can be attached to UW 'partner(ic>...)' . Because if "A is a partner of B" is true, "B is a partner of A" will be true also.
- '@identifiable' can be attached to (compound) UW 'national health insurance id' as the content of the UW can identify the person who hold the ID.
- '@disjointed' can be attached to a scope consists of "men and women" as these two element concepts do not share common instances.

Relations for expressing logical relations

These relations of UNL have been developed to focus on expressing semantics of a sentence. From this point of view, the relations can be divided into three groups: 1) to express case relations (agt, obj, etc), 2) to express event relations (con, seq, etc), and 3) to express conceptual relations (cnt, euq, etc). Table 9 and 10 shows the relations to express logical relations.

Table 9: relations that can be used to express logical relations between concepts

and	Connects two concepts that values are true. Corresponds to the logical operator "union".
-----	--

con	Shows the conditional concepts. Corresponds to the logical operator "imply" but the direction is reverse.
int	Connects two concepts and takes an intersection. Corresponds to the logical operator "intersection"
or	Connects two events or states that one of them is true at least. Corresponds to the logical operator "exclusive union"

Table 10: relations that can express logical relations

cnt	Shows the content of a concept
equ	Link to a equivalent concept
icl	Link to a upper class concept
iof	Link an instance to a class concept

Conclusion

The UNL is a language for computer to express information and knowledge that can be described in natural languages. Methodology of defining UWs allows to express any specific concept of a particular language precisely without ambiguities and understandable for computers and for humans of other languages. This mechanism allow all languages to have interfaces with UNL and makes it possible that anyone can participate in the development of a language module or UNL-based applications. Information and knowledge that UNL will treat in the future needs a wide range of people to develop them. The knowledge expressed by UNL are universal and can be shared by all nations.

With UNL, knowledge can be expressed understandable to computers. This means that computers can reason using the knowledge. The UNL is a language for providing the Knowledge Infrastructure for more intelligent processing such as semantic computing. The UNL expresses knowledge provided only by natural languages at this moment. Whereas CDL (Conceptual Description Language: an integration of [UNL](#) (Universal Networking Language), RDF, OWL, and so forth) is aimed at dealing with contents provided by all kinds of media

Related materials and References

- [1] UNDL Foundation : UNL System <http://www.undl.org/unlsys/>
- [2] UNDL Foundation: UNL Specifications 2005 <http://www.undl.org/unlsys/unl/UNL2005/>, to be available in May 2005
- [3] UNDL Foundation: UNL Specifications v3.3 <http://www.undl.org/unlsys/unl/UNLSpecs33.pdf>, 2004
- [4] UNDL Foundation: UNL explorer and UNL Encyclopedia <http://www.undl.org/encyclopedia/>, 2004
- [5] Kinsui Satoshi and Ikumi Imani: Semantics and Context, Iwanami shoten, Introduction to contemporary linguistics 4, 2003