Interdisciplinary Journal of Linguistics
Volume [8] 2015, Pp.189-197

Analysis - Grammar in UNL: An Initiative towards Machine Translation for Kashmiri

Sumaira Nabi
Sumaya Jehangir *

Abstract

Universal Networking Language (UNL) is a computer language created to represent and process information across language barriers (Uchida et al, 2001). The two basic tools in the UNL system are Enconverter and Deconverter. Enconverter is a language independent parser that converts NL sentence into UNL and Deconverter is a language independent generator that can convert UNL expression to a variety of languages, using respective language's Grammatical rules and Dictionary. The present work is an attempt to use UNL for the purpose of MT. The paper presents the Analysis rules and Dictionary for some structures of Kashmiri. The derived UNLised text can be deconverted to any language by the Deconverter tool. The usefulness of Interlingua- based MT system over other MT systems has also been discussed in this paper.

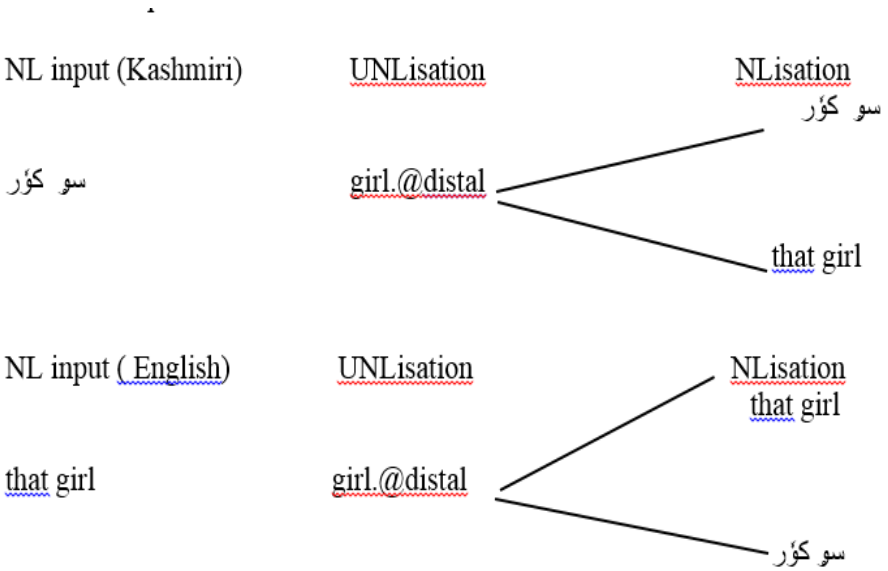
Keywords: UNL, MT, Enconverter, Deconverter, UNL expression

Introduction:

Machine translation (MT) is nowadays one of the most emerging fields in computational linguistics. MT is the automatic translation from one natural language to another done by a machine. The area of MT has witnessed integration of research work from other fields like statistics, mathematics, artificial intelligence and many others during its expansion. Different approaches like Direct MT, Rule Based MT, corpus based MT and knowledge based MT have been proposed for the purpose of Machine translation. Universal Networking Language (UNL) based MT working on Interlingual approach is also an effort in this direction. Universal Networking Language (UNL) is a computer language created to represent and process information across language barriers (Uchida et al, 2001). The UNL programme was launched in 1996 in the Institute of Advanced studies (IAS) of United Nations University (UNU), Tokyo, Japan.

*Department of Linguistics, University of Kashmir.

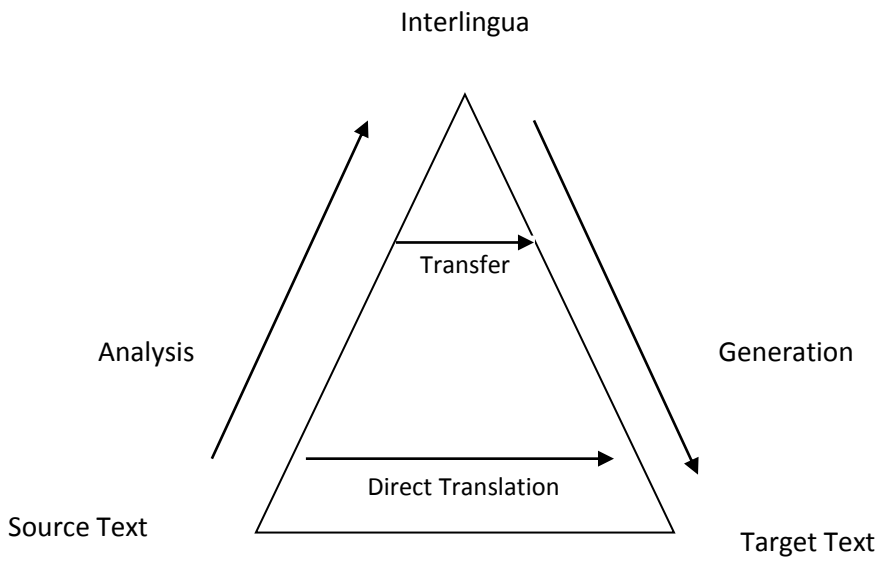
UNL represents Natural Language in an unambiguous format.UNL representations explicitly codify all the information present in NL.



From the above example, it is clear that UNLised text can be converted to any language.The overall architecture of the UNL system has been developed with a set of basic software and tools necessary for its functioning. UNL programme centers around the development of an Enconverter and a Deconverter for a natural language. Enconverter converts natural language text to an equivalent UNL expression and the Deconverterdeconverts a given UNL expression to an equivalent natural language sentence.

Interlingua Based MT and other MT systems:

MT systems based on Interlingual approach are most appropriate for multilingual machine MT, as these systems require only two components for each language .The comparison of Direct MT, Transfer based MT and Interlingual MT can be explained with the help of Vacquois triangle.



Vacquois Traingle

Vacquois triangle compares the features of Direct MT, Transfer based MT and Interlinual based MT systems. The Vacquois triangle was used in the linguistic rule based era of machine translation to describe the complexity of approaches to MT and also the evolution of these approaches. The first approach used was a direct word for word substitution between languages. Later efforts moved up the pyramid and introduced more complex processing, and also a modularization of the process into steps, beginning with analysis of the source language ,transfer of information between the languages, and then generation of target language output. The Interlingua systems analyse the source language text into an abstract universal language- independent representation of meaning (the Interlingua), and generating this meaning using the lexical units and the syntactic constructions of the target language.The Interlingua approach appears at the apex of the Vauquois triangle. It is clear from the above fig. that from base to apex MT requires greater effort in source language analysis and target language generation but reduces the effort involved in conversion between languages.

Representation Of Information In UNL:

In the UNL programme, information extracted from natural language text is represented in the form of a semantic network (UNL graph), composed of three distinct semantic entities i.e. universal words, relations and attributes.

UW's are the nodes of the UNL graph and can represent simple or compound concepts.

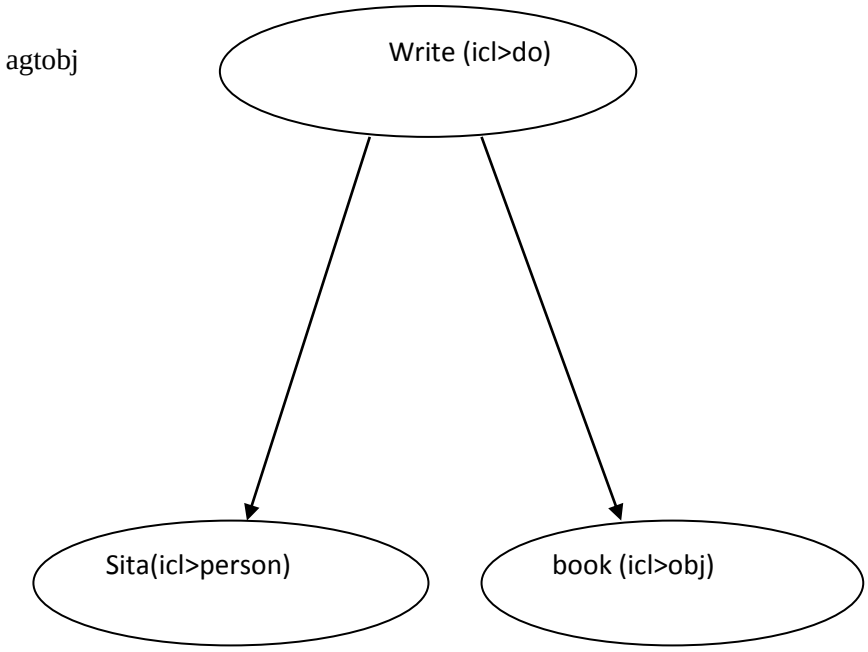
Attribute labels express additional information about the UW's that appear in a sentence. It includes tense, number, aspect, represent information on the role of the node in the UNL graph as in the case of @entry that indicates the main node of the graph.

Relations formerly known as "links" are labeled arcs connecting a node to another node in a UNL graph (Martins, 2002). Consider the example below:

سیتا چھے لِکھان کِتاب

Si:**ta**:	**tʃe**	**lekha**:**n**	**Kita**:**b**
sita	be-PR	write-CONT	book

@entry.@present@progress

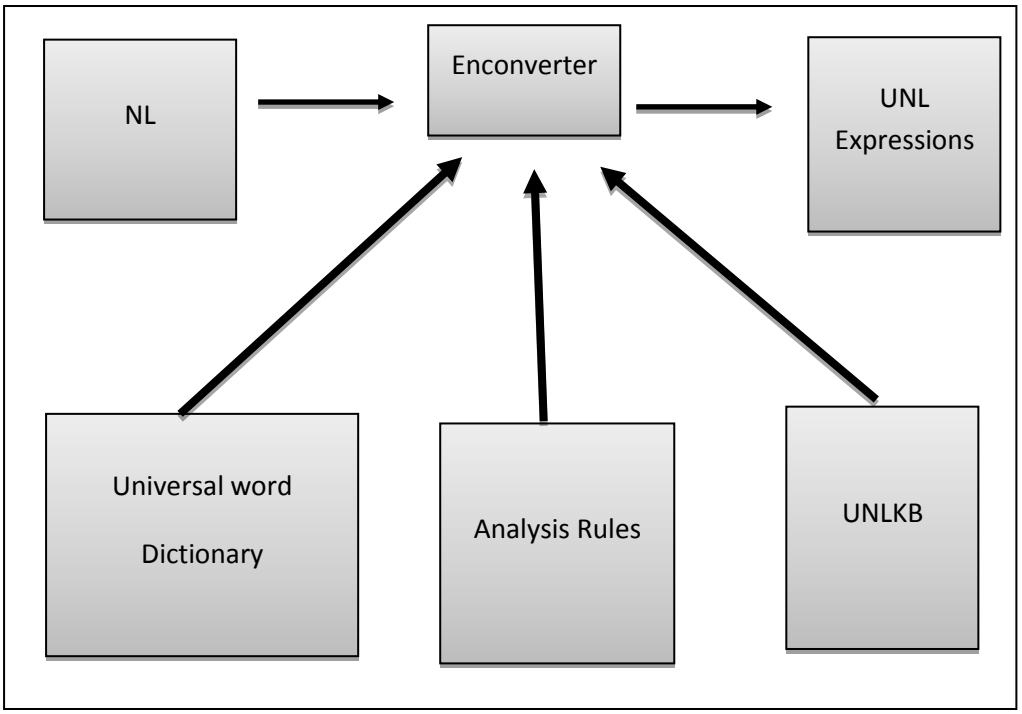


The above figure represents UNL representation of سیتا چھے لِکھان کِتاب

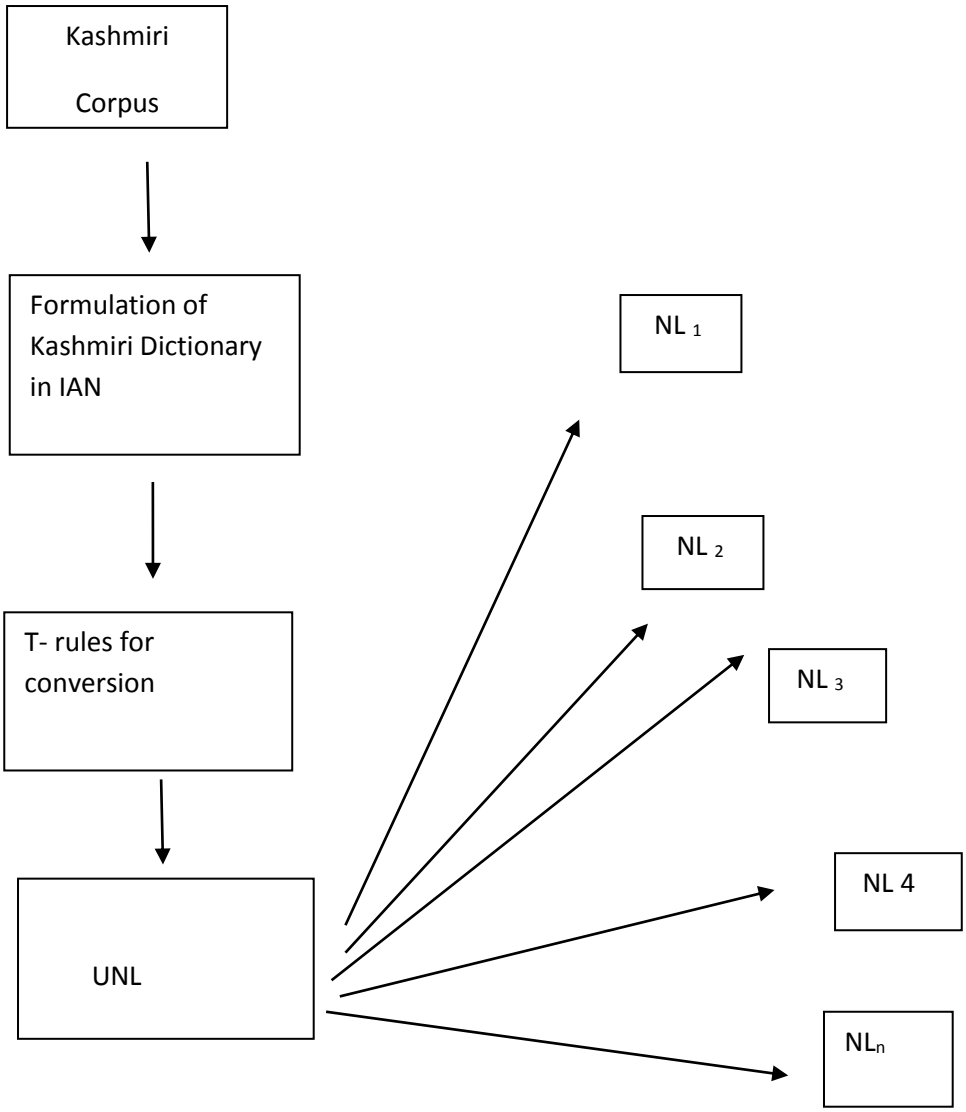
In this graph "write (icl>do)", sita (iof>person) and book (icl>obj) are UW's. agt (agent) and obj (object) are relations and @entry.@present.@progress are attributes.

UNL Representation:
[UNL]
agt(write(icl>do)@entry.@present.@progress, sita(iof>person))
obj(write(icl>do)@entry.@present.@progress,book(icl>object))
[/UNL]

Components of Enconverter:



Stepwise Enconversion scheme for Kashmiri corpus:



The above fig. shows the stepwise scheme for UNLisation. The UNLised text can be converted to any language with its own Deconverter. Enconverter is a language independent software applicable to any language. Enconverter converts Natural Language text into UNL. The process of Enconversion requires a word dictionary and Analysis Grammar.

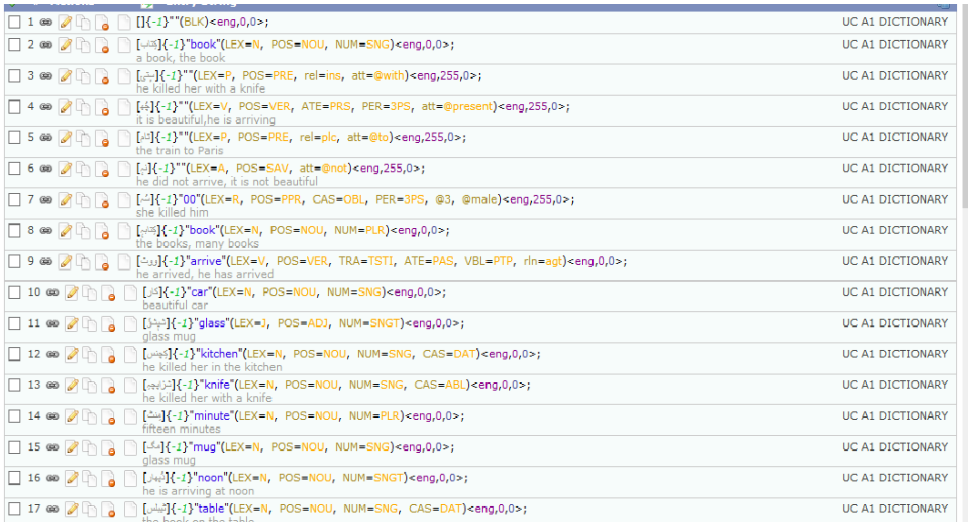
The format of Dictionary entries in IAN is as:

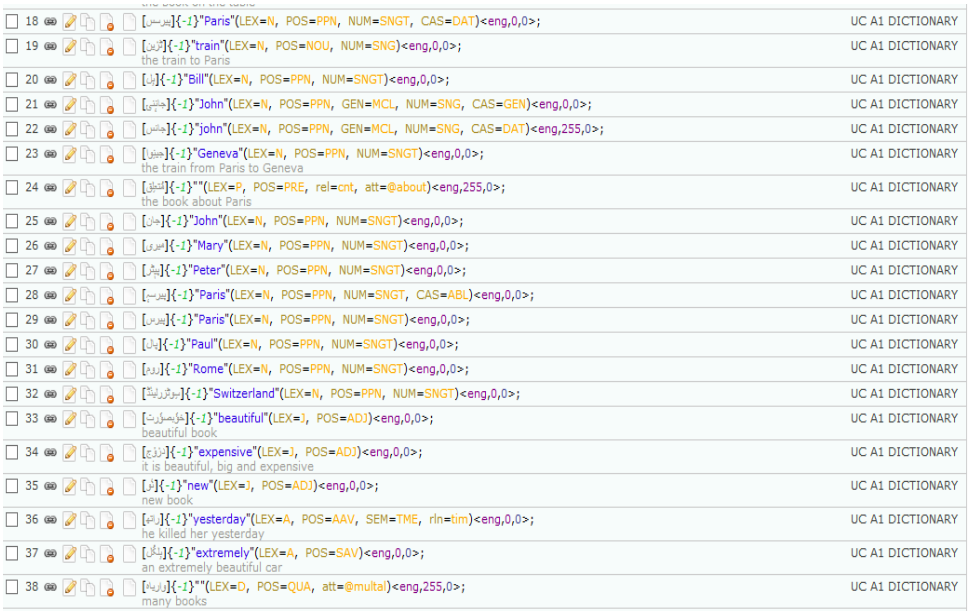
[HW]{ID}"UW" (ATTR….) <FLG,FRE);

Where,
HW= Head Word
FLG= Language Flag
FRE=Frequency of Head Word

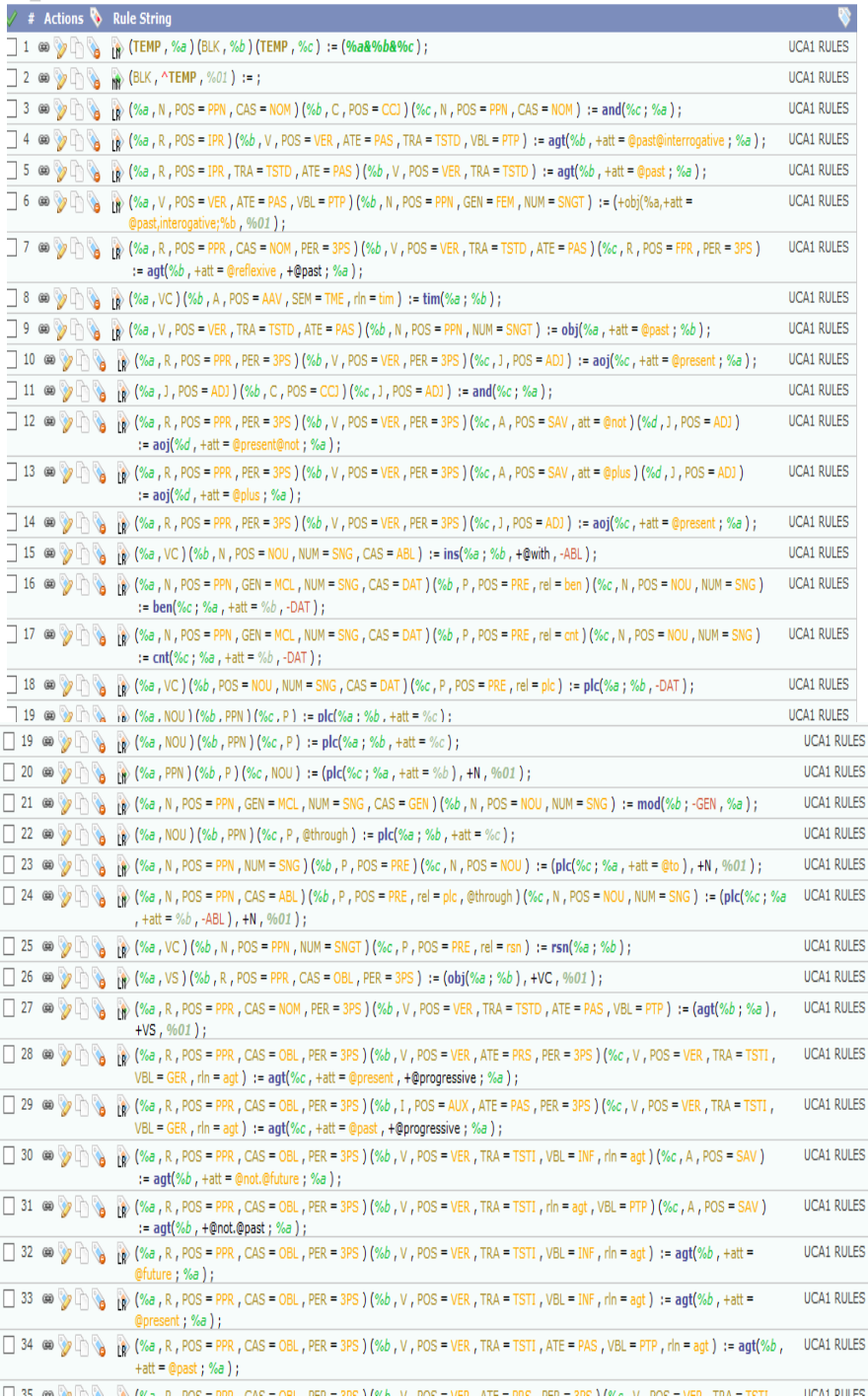
The screenshots of the Dictionary and Enconversion Rules (Analysis Grammar) framed for Kashmiri corpus in IAN (Interactive Analyzer) are shown below:

Dictionary Entries in IAN





T-Rules in IAN



#	Actions	Rule String	
1		(TEMP , %a) (BLK , %b) (TEMP , %c) := (%a&%b&%c) ;	UCA1 RULES
2		(BLK , ^TEMP , %01) := ;	UCA1 RULES
3		(%a , N , POS = PPN , CAS = NOM) (%b , C , POS = CCJ) (%c , N , POS = PPN , CAS = NOM) := and(%c ; %a) ;	UCA1 RULES
4		(%a , R , POS = IPR) (%b , V , POS = VER , ATE = PAS , TRA = TSTD , VBL = PTP) := agt(%b , +att = @past@interrogative ; %a) ;	UCA1 RULES
5		(%a , R , POS = IPR , TRA = TSTD , ATE = PAS) (%b , V , POS = VER , TRA = TSTD) := agt(%b , +att = @past ; %a) ;	UCA1 RULES
6		(%a , V , POS = VER , ATE = PAS , VBL = PTP) (%b , N , POS = PPN , GEN = FEM , NUM = SNGT) := (+obj(%a,+att = @past,interogative;%b , %01) ;	UCA1 RULES
7		(%a , R , POS = PPR , CAS = NOM , PER = 3PS) (%b , V , POS = VER , TRA = TSTD , ATE = PAS) (%c , R , POS = FPR , PER = 3PS) := agt(%b , +att = @reflexive , +@past ; %a) ;	UCA1 RULES
8		(%a , VC) (%b , A , POS = AAV , SEM = TME , rln = tim) := tim(%a ; %b) ;	UCA1 RULES
9		(%a , V , POS = VER , TRA = TSTD , ATE = PAS) (%b , N , POS = PPN , NUM = SNGT) := obj(%a , +att = @past ; %b) ;	UCA1 RULES
10		(%a , R , POS = PPR , PER = 3PS) (%b , V , POS = VER , PER = 3PS) (%c , J , POS = ADJ) := aoj(%c , +att = @present ; %a) ;	UCA1 RULES
11		(%a , J , POS = ADJ) (%b , C , POS = CCJ) (%c , J , POS = ADJ) := and(%c ; %a) ;	UCA1 RULES
12		(%a , R , POS = PPR , PER = 3PS) (%b , V , POS = VER , PER = 3PS) (%c , A , POS = SAV , att = @not) (%d , J , POS = ADJ) := aoj(%d , +att = @present@not ; %a) ;	UCA1 RULES
13		(%a , R , POS = PPR , PER = 3PS) (%b , V , POS = VER , PER = 3PS) (%c , A , POS = SAV , att = @plus) (%d , J , POS = ADJ) := aoj(%d , +att = @plus ; %a) ;	UCA1 RULES
14		(%a , R , POS = PPR , PER = 3PS) (%b , V , POS = VER , PER = 3PS) (%c , J , POS = ADJ) := aoj(%c , +att = @present ; %a) ;	UCA1 RULES
15		(%a , VC) (%b , N , POS = NOU , NUM = SNG , CAS = ABL) := ins(%a ; %b , +@with , -ABL) ;	UCA1 RULES
16		(%a , N , POS = PPN , GEN = MCL , NUM = SNG , CAS = DAT) (%b , P , POS = PRE , rel = ben) (%c , N , POS = NOU , NUM = SNG) := ben(%c ; %a , +att = %b , -DAT) ;	UCA1 RULES
17		(%a , N , POS = PPN , GEN = MCL , NUM = SNG , CAS = DAT) (%b , P , POS = PRE , rel = cnt) (%c , N , POS = NOU , NUM = SNG) := cnt(%c ; %a , +att = %b , -DAT) ;	UCA1 RULES
18		(%a , VC) (%b , POS = NOU , NUM = SNG , CAS = DAT) (%c , P , POS = PRE , rel = plc) := plc(%a ; %b , -DAT) ;	UCA1 RULES
19		(%a , NOU) (%b , PPN) (%c , P) := plc(%a ; %b , +att = %c) ;	UCA1 RULES
19		(%a , NOU) (%b , PPN) (%c , P) := plc(%a ; %b , +att = %c) ;	UCA1 RULES
20		(%a , PPN) (%b , P) (%c , NOU) := (plc(%c ; %a , +att = %b) , +N , %01) ;	UCA1 RULES
21		(%a , N , POS = PPN , GEN = MCL , NUM = SNG , CAS = GEN) (%b , N , POS = NOU , NUM = SNG) := mod(%b ; -GEN , %a) ;	UCA1 RULES
22		(%a , NOU) (%b , PPN) (%c , P , @through) := plc(%a ; %b , +att = %c) ;	UCA1 RULES
23		(%a , N , POS = PPN , NUM = SNG) (%b , P , POS = PRE) (%c , N , POS = NOU) := (plc(%c ; %a , +att = @to) , +N , %01) ;	UCA1 RULES
24		(%a , N , POS = PPN , CAS = ABL) (%b , P , POS = PRE , rel = plc , @through) (%c , N , POS = NOU , NUM = SNG) := (plc(%c ; %a , +att = %b , -ABL) , +N , %01) ;	UCA1 RULES
25		(%a , VC) (%b , N , POS = PPN , NUM = SNGT) (%c , P , POS = PRE , rel = rsn) := rsn(%a ; %b) ;	UCA1 RULES
26		(%a , VS) (%b , R , POS = PPR , CAS = OBL , PER = 3PS) := (obj(%a ; %b) , +VC , %01) ;	UCA1 RULES
27		(%a , R , POS = PPR , CAS = NOM , PER = 3PS) (%b , V , POS = VER , TRA = TSTD , ATE = PAS , VBL = PTP) := (agt(%b ; %a) , +VS , %01) ;	UCA1 RULES
28		(%a , R , POS = PPR , CAS = OBL , PER = 3PS) (%b , V , POS = VER , ATE = PRS , PER = 3PS) (%c , V , POS = VER , TRA = TSTI , VBL = GER , rln = agt) := agt(%c , +att = @present , +@progressive ; %a) ;	UCA1 RULES
29		(%a , R , POS = PPR , CAS = OBL , PER = 3PS) (%b , I , POS = AUX , ATE = PAS , PER = 3PS) (%c , V , POS = VER , TRA = TSTI , VBL = GER , rln = agt) := agt(%c , +att = @past , +@progressive ; %a) ;	UCA1 RULES
30		(%a , R , POS = PPR , CAS = OBL , PER = 3PS) (%b , V , POS = VER , TRA = TSTI , VBL = INF , rln = agt) (%c , A , POS = SAV) := agt(%b , +att = @not.@future ; %a) ;	UCA1 RULES
31		(%a , R , POS = PPR , CAS = OBL , PER = 3PS) (%b , V , POS = VER , TRA = TSTI , rln = agt , VBL = PTP) (%c , A , POS = SAV) := agt(%b , +@not.@past ; %a) ;	UCA1 RULES
32		(%a , R , POS = PPR , CAS = OBL , PER = 3PS) (%b , V , POS = VER , TRA = TSTI , VBL = INF , rln = agt) := agt(%b , +att = @future ; %a) ;	UCA1 RULES
33		(%a , R , POS = PPR , CAS = OBL , PER = 3PS) (%b , V , POS = VER , TRA = TSTI , VBL = INF , rln = agt) := agt(%b , +att = @present ; %a) ;	UCA1 RULES
34		(%a , R , POS = PPR , CAS = OBL , PER = 3PS) (%b , V , POS = VER , TRA = TSTI , ATE = PAS , VBL = PTP , rln = agt) := agt(%b , +att = @past ; %a) ;	UCA1 RULES

Conclusion

The present work provided the input to the Enconverter in UNL for the UNLisation of some selected corpus from Kashmiri. The generated UNL expressions can be deconverted to a variety of languages by the Deconverter tool. Till date, very little has been done regarding the development of MT for kashmiri, and it is expected that this work will serve as an initial step towards the development of an online MT system for Kashmiri.UNL based MT systems translate the text with an appreciable accuracy. Moreover, these sytems are comparatively time and cost efficient.

References

Cardenosa, Jesús, Alexander Gelbukh, and Edmundo Tovar (Eds) 2005. *Universal Networking Language: Advances in Theory and Application.* InstitutoPolitecnicoNacional, Centro de Investigation e. computation: Mexico.

Cardeñosa, Jesús, Carolina Gallardo Pérez, Adriana Toni, 2009. "Multilingual Cross Language Information Retrieval: A New Approach". *Universal Networking Language: Present Status and Perspectives Workshop, in conjunction with CSIT 7th International Conference.*

*Yerevan.*http://www.vai.dia.fi.upm.es/publicaciones/multi_draft_csit09.pdf

Prashanth. K., 2010 web. *Semantics Extraction from Text*. Diss. Indian Institute of Technology,

Bombay.<http://www.cse.iitb.ac.in/alumni/~pkamle08/website/docs/thesis.pdf>

Martins, Ronaldo, 2002 web. *UNL Developers' Guide*. Vol. 2. UNDL Foundation: Geneva

< http://www.ronaldomartins.pro.br/unlx/Attributes.pdf>

Uchida, H. and M. Zhu, 1998. "The Universal Networking Language (UNL) Specification Version 3.0". *Technical Report, United Nations University, Tokyo.*Uchida, H., M. Zhu and T.C.D. Senta, 2005. *Universal Networking Language*. UNDL Foundation: Geneva.

UNLweb. 2010. *UNDL Foundation*. http://www.unlweb.net/unlweb/

❐❐❐