The ICON-2010 tools contest on Indian language dependency parsing

Samar Husain, Prashanth Mannem, Bharat Ambati and Phani Gadde
Language Technologies Research Centre, IIIT-Hyderabad, India.
{samar, prashanth, ambati, phani.gadde}@research.iiit.ac.in

Abstract

The ICON10 tools contest was dedicated to the task of dependency parsing for Indian languages (IL). Three languages namely, Hindi, Telugu and Bangla, were explored. The motivation behind the task was to investigate and solve the challenges in IL parsing by making annotated data available to the larger community.

1 Introduction

The tools contest at International Conference on Natural Language Processing (ICON) is a regular event that aims at building/improving Indian language (IL) NLP tools. Following the enthusiastic response of ICON09 contest on IL dependency parsing (Husain, 2009), a follow-up contest on the same topic was organized. Husain (2009) describes the participating systems. Crucial parsing issues, many IL specific, came to light and were discussed. However, efficient Indian Language (IL) parsing still remains a challenging task. Most Indian languages are morphologically rich and free word order (MoR-FWO). It is known that MoR-FWO languages pose various challenges for the task of parsing because of their non-configurationality. Also, the syntactic cues necessary to identify various relations in such languages are complex and distributed. This problem worsens in the context of data-driven dependency parsing due to non-availability of large annotated corpus. Past experiments on parser evaluation and parser adaptation for MoR-FWO languages (like Turkish, Basque, Czech, Arabic, Hebrew, etc.) have shown that there are a number of factors which contribute to the performance of a parser (Nivre et al., 2007b; Hall et al. 2007; McDonald and Nivre, 2007). For Hindi, (a) *difficulty in extracting relevant linguistic cues*, (b) *non-projectivity*, (c) *lack of explicit cues*, (d) *long distance dependencies*, (e) *complex linguistic phenomena*, and (f) *small corpus size*, have been suggested as possible reasons for low performance (Bharati et al., 2008, Ambati et al., 2010a). There has been a recent surge in addressing parsing for MoR-FOW languages (Nivre and McDonald, 2008; Nivre, 2009; Tsarfaty and Sima'an, 2008; Seddah et al., 2009; Gadde et al., 2010; Husain et al., 2009, Eryigit et al., 2008; Goldberg and Elhedad, 2009, Tsafarty et al., 2010; Mannem et al., 2009). It is our hope that the ICON10 tools contest will add to this knowledge.

2 Annotated Data

The data for all the three languages was annotated using the Computational Paninian Grammar (Bharati et al., 1995). The annotation scheme based on this grammar has been described in Begum et al. (2008) and Bharati et al. (2009a). Table 1 shows the training, development and the testing data sizes for the Hindi, Telugu and Bangla Treebanks.

Type	Lang.	Sent Count	Word Count	Avg. sent_length
	Hindi	2,972	64632	22.69
Train	Telugu	1,400	7602	5.43
	Bangla	980	10305	10.52
	Hindi	543	12617	23.28
Devel	Telugu	150	839	5.59
	Bangla	150	1196	7.97
	Hindi	320	6589	20.59
Test	Telugu	150	836	5.57
	Bangla	150	1350	9.0

Table 1. Treebank Statistics

2.1 Dependency Tagset

The tagset used in the dependency framework is syntactic-semantic in nature and has 59 labels. Keeping in mind the small size of the current treebanks an additional coarse-grained tagset containing 37 labels was derived from the original tagset. The coarse-grained tagset data was automatically created from the original treebank by mapping the original tagset to the coarse-grained tagset. Hence, two sets of data for each language were released. APPENDIX-I shows the original tagset (henceforth referred as the fine-grained tagset). The mapping from fine to coarse tags is shown in APPENDIX-I.

2.2 Information in the released data

The released annotated data for the three languages has the following information:

(1) Morphological information
(2) Part of Speech (POS) tag
(3) Chunk boundary and chunk tag
(4) Chunk head information
(5) *Vibhakti*[1] of the head
(6) Dependency relation

Morph output has the following information

a) *Root*: Root form of the word
b) *Category*: Course grained POS·
c) *Gender*: Masculine/Feminine/Neuter
d) *Number*: Singular/Plural
e) *Person*: First/Second/Third person
f) *Case*: Oblique/Direct case
g) *Vibhakti*: Vibhakti of the word

POS and chunk annotation follows the scheme proposed by Bharati et al. (2006b). The dependency annotation for all the three languages was done on chunks. A chunk groups local dependencies that have no effect on the global dependency structure. In general, all the nominal inflections, nominal modifications (adjective modifying a noun, etc.) are treated as part of a noun chunk, similarly, verbal inflections, auxiliaries are treated as part of the verb chunk (Bharati et al., 2006b). For each sentence in the corpus, the POS tags and the chunks along with their head words are marked first and in the next step, dependencies are annotated between these chunk heads. The dependency relations between words within a chunk are not marked and they can be derived automatically.

Due to this scheme of annotation, the treebanks contain only inter-chunk relations. For Hindi, an automatic tool was used to expand the chunks to get the intra-chunk relations (Bharati etl al, 2009b). The performance of this tool is close to 96%. Due to unavailability of such a tool for Telugu and Bangla, the dependency relations in these treebanks are between chunk heads only.

Table 2 shows how all the above information has been marked in the treebanks. For Hindi, the morph features, POS tags, chunk tags and boundaries along with inter-chunk dependencies have been manually annotated. The head word of the chunk, its vibhkati and the intra-chunk dependencies were automatically marked. For Telugu and Bangla, only POS, chunk and inter-chunk dependency information are manually annotated. This holds true for training, development and the test data.

Lang	POS	Ch	Dep	Mo	Head	Vib.
Hin	Man	Man	Man/ Auto	Man	Auto	Auto
Tel	Man	Man	Man	Auto	Auto	Auto
Ban	Man	Man	Man	Auto	Auto	Auto

Table 2. Lang: Language, Hin: Hindi, Tel: Telugu, Ban: Bangla POS: POS tags, Ch: Chunk boundaries and tags, Dep: Dependency relations, Mo: Morphological features, Head: Chunk head information, Vib: *vibhakti* of head. Man: Manual, Auto: Automatic

3 Contest

3.1 Data format

The annotation for all the three treebanks were done in Shakti Standard Format (SSF) (Bharati et al., 2006a). SSF allows for a multi-layered representation in a single data structure. Each sentence is wrapped in a XML tag to indicate the start and end of a sentence. For each sentence, the word id, word form/chunk boundary, POS/chunk tag and features are listed in four columns respectively. The features column contains the morphological and dependency information.

Since CoNLL-X is a widely used representation for dependency parsers, hence the treebanks for all the three langauges were released in SSF as well as in CoNLL-X format.

As Telugu and Bangla treebanks do not have the intra-chunk relations, the sentences in the CoNLL-X data for these languages contain just the chunk heads. The non-head of a chunk are absent in the CoNLL-X format. However, the post-positions and auxiliary verbs (crucial for parsing ILs) which normally are non-heads in a chunk are listed as chunk head features.. The SSF data, on the other hand, retains the full sentences.

3.2 Evaluation

The standard dependency evaluation metrics like Unlabeled Attachment Score (UAS), Label Accuracy (LA), and Labeled Attachment Score (LAS) have been used to evaluate the submis-

[1] *Vibhakti* is a generic term for nominal case-marking, post-positions and verbal inflections, tense, aspect , modality.

sions (Nivre et al., 2007a). UAS is the percentage of words in the sentences across the entire test data that have correct parents. LA is the percentage of words with correct dependency label, while LAS is the percentage of words with correct parent and correct dependency label.

For evaluation on the test data, the teams were asked to submit two separate outputs with coarse grained and fine grained dependency labels. The three scores were computed for both the outputs.

4 Submissions

All the participating teams were expected to submit their system's results for both fine-grained and coarse-grained tagset data. In all, around 15 teams registered for the contest. 6 teams eventually submitted the results. Out of these, 4 teams submitted outputs for all the three languages and 2 teams did it only for one language.

4.1 Approaches

Attardi et al. (2010) use a transition based dependency shift reduce parser (DeSR parser) that uses a Multilayer Perceptron (MLP) classifier with a beam search strategy. They explore the effectiveness of morph agreement and chunk features by stacking MLP parses obtained from different configurations.

Ghosh et al. (2010) describe a dependency parsing system for Bengali in which MaltParser is used as a baseline system and its output is corrected using a set of post-processing rules.

Kosaraju et al. (2010) use MaltParser and explored the effectiveness of local morpho-syntactic features, chunk features and automatic semantic information. Parser settings in terms of different algorithms and features were also explored.

Abhilash and Mannem (2010) use a bidirectional parser with perceptron learning for all the three languages with rich context as features.

Kesedi et al. (2010) use a hybrid constraint based parser for Telugu. The scoring function for ranking the base parses is inspired by a graph-based parsing model and labeling.

Kolachina et al. (2010) use MaltParser to explore the effect of valency and arc-directionality on the parser performance. They employ feature propagation in order to incorporate such features during the parsing process. They also explore various blended systems to get the best accuracy.

Team	Approach	Learning Algorithm
Attardi	DeSR	Multilayer Perceptron
Abhilash	Bidirectional Parsing	Perceptron
Ghosh	MaltParser + Post processing	SVM
Kesedi	Constraint based hybrid Parsing	MaxEnt
Kolachina	MaltParser	SVM
Kosaraju	MaltParser	SVM

Table 3. Systems summary

4.1.1 Labeling

Labeling dependencies is a hard problem in IL dependency parsing due to the non-configurational nature of these languages. Ambati et al. (2010b) attribute this to, among others, the absence of postpositions and ambiguous postpositions.

Abhilash and Mannem (2010) used the bidirectional parser for unlabeled parsing and assigned labels in the next stage using a maximum entropy classifier. The constraint system of Kesedi et al. (2010) uses manually created grammar to get the labels. A maximum entropy labeler is then used to select the best parse. Teams which used MaltParser did unlabeled and labeled parsing together (Kolachina et al., 2010; Kosaraju et al., 2010). Ghosh et al. (2010) too used labels from MaltParser, but these labels were corrected in some cases using rules in a post-processing step. Attardi et al. (2010) too performed labeling along with unlabeled dependency parsing.

4.1.2 Features

Apart from the telugu constraint based parser (Kesedi et al, 2010), all the other teams used state of the art data driven dependency parsers which have been proposed earlier. And therefore, the features used in these systems become the crucial differentiating factor. In what follows we point out the novel linguistically motivated features used by the teams.

Attardi et al. (2010) use morph agreement as feature along with chunk features. For time and location dependency labels, word lists of dependents are collected during training to be used while parsing.

Kolachina et al. (2010) experiment with dependency arc direction and valency through feature propagation. Valency information is captured using the dependency labels of the outgoing arcs which are deemed mandatory for the parent node.

Kosaraju et al. (2010) successfully use features derived from a shallow parser's output for Hindi. Semantic features such as *human*, *nonhuman*, *inanimate* and *rest* are used for parsing without much success, mainly due to the difficulty in correctly predicting the semantic features for a new sentence.

It would be interesting to see the effect of all these novel features (morph agreement, valency information, semantic labels) in a single parser.

5 Results

Evaluation is done on system's output for both fine-grained and coarse-grained tagset data of all the three languages. Average of all the six (3 languages * 2 tagsets) outputs is taken for final evaluation. Out of 6 teams, 4 teams submitted all the six outputs. Gosh et al. (2010) submitted output for fine-grained Bangla data and Kesedi et al. (2010) submitted output for coarse-grained Telugu data. Best average UAS of 90.94% is achieved by Attardi et al. (2010). Whereas, best average LAS of 76.83% and LS of 79.14 is achieved by Kosaraju et al. (2010). Table 4 (a-c) shows the consolidated results for all the system.

6 Conclusion

Many new parsing approaches were successfully adapted for ILs in ICON10 tools contest. Previously unexplored features were also investigated. The state-of-the-art parsing accuracies for Hindi and Telugu were improved.

The large jump in the Hindi accuracies were mainly due to the high precision in the identification of intra-chunk relations. This is because, intra-chunk relations are mostly local syntactic dependencies that can be easily identified (Ambati et al., 2010b). The chunk level dependency issues pointed out during the ICON 2009 tools contest on IL dependency parsing (Husain, 2009), as well as in Bharati et al. (2008) and Ambati et al. (2010a) that lead to the low LAS in IL parsing still remain pertinent. This means that there remains a lot of scope for improvement, especially, in bridging the gap between UAS and LAS.

References

A. Abhilash and P. Mannem. 2010. Bidirectional Dependency Parser for Indian Languages. In Proc of *ICON-2010 tools contest on Indian language dependency parsing*. Kharagpur, India.

B. Ambati, S. Husain, J. Nivre and R. Sangal. 2010a. On the Role of Morphosyntactic Features in Hindi Dependency Parsing. *In Proceedings of NAACL-HLT 2010 workshop on Statistical Parsing of Morphologically Rich Languages (SPMRL 2010), Los Angeles, CA.*

B. Ambati, S. Husain, S. Jain, D. M. Sharma and R. Sangal. 2010b. Two methods to incorporate 'local morphosyntactic' features in Hindi dependency parsing. *In Proceedings of NAACL-HLT2010 workshop on Statistical Parsing of Morphologically Rich Languages (SPMRL 2010) LosAngeles, CA.*

G. Attardi, S. D. Rossi and M. Simi. 2010. Dependency Parsing of Indian Languages with DeSR. In Proc of *ICON-2010 tools contest on Indian language dependency parsing*. Kharagpur, India.

R. Begum, S. Husain, A. Dhwaj, D. M. Sharma, L. Bai and R. Sangal. 2008. Dependency Annotation Scheme for Indian Languages. *In Proceedings of The Third International Joint Conference on Natural Language Processing (IJCNLP). Hyderabad, India.*

A. Bharati, D. M. Sharma, S. Husain, L. Bai, R. Begam and R. Sangal. 2009a. *AnnCorra:TreeBanks for Indian Languages, Guidelines for Annotating Hindi Treebank***.** http://ltrc.iiit.ac.in/MachineTrans/research/tb/DS-guidelines/DS-guidelines-ver2-28-05-09.pdf

A. Bharati, S. Husain, D. M. Sharma, L. Bai, and R. Sangal. 2009b. *AnnCorra:TreeBanks for Indian Languages, Guidelines for Annotating Intra-chunk dependency***.** http://ltrc.iiit.ac.in/MachineTrans/research/tb/intra-chunk%20guidelines.pdf

A. Bharati, S. Husain, B. Ambati, S. Jain, D. M. Sharma and R. Sangal. 2008. Two semantic features make all the difference in Parsing accuracy. *In Proceedings of the 6th International Conference on Natural Language Processing (ICON-08), DAC Pune, India.*

A. Bharati, R. Sangal and D. M. Sharma. 2006a. SSF: Shakti Standard Format Guide. *LTRC-TR33*. http://ltrc.iiit.ac.in/MachineTrans/publications/technicalReports/tr033/SSF.pdf

A. Bharati, D. M. Sharma, L. Bai and R. Sangal. 2006b. AnnCorra: Annotating Corpora Guidelines for POS and Chunk Annotation for Indian Languages. *LTRC-TR31*. http://ltrc.iiit.ac.in/MachineTrans/publications/technicalReports/tr031/posguidelines.pdf

A. Bharati, V. Chaitanya, R. Sangal. 1995. *Natural Language Processing: A Paninian Perspective.* Prentice-Hall of India, New Delhi.

P. Gadde, K. Jindal, S. Husain, D. M Sharma, and R. Sangal. 2010. Improving Data Driven Dependency Parsing using Clausal Information. *In Proceedings of NAACL-HLT 2010, Los Angeles, CA. 2010.*

A. Ghosh, A. Das, P. Bhaskar and S. Bandyopadhyay. 2010. Bengali Parsing System at ICON NLP Tool Contest 2010. In Proc of *ICON-2010 tools contest on Indian language dependency parsing*. Kharagpur, India.

Y. Goldberg and M. Elhadad. 2009. Hebrew Dependency Parsing: Initial Results. *In Proceedings of the 11th IWPT09. Paris. 2009.*

J. Hall, J. Nilsson, J. Nivre, G. Eryigit, B. Megyesi, M. Nilsson and M. Saers. 2007. Single Malt or Blended? A Study in Multilingual Parser Optimization. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007,* 933—939

S. Husain, P. Gadde, B. Ambati, D. M. Sharma and R. Sangal. 2009. A modular cascaded approach to complete parsing. *In Proceedings of the COLIPS International Conference on Asian Language Processing 2009 (IALP). Singapore.*

S. Husain. 2009. Dependency Parsers for Indian Languages. *In Proceedings of ICON09 NLP Tools Contest: Indian Language Dependency Parsing. Hyderabad, India. 2009.*

S. R. Kesidi, P. Kosaraju, M. Vijay and S. Husain. 2010. A Two Stage Constraint Based Hybrid Dependency Parser for Telugu. In Proc of *ICON-2010 tools contest on Indian language dependency parsing*. Kharagpur, India.

S. Kolachina, P. Kolachina M. Agarwal, and S. Husain. 2010. Experiments with MaltParser for parsing Indian Languages. In Proc of *ICON-2010 tools contest on Indian language dependency parsing*. Kharagpur, India.

P. Kosaraju, S. R. Kesidi, V. B. R. Ainavolu and P. Kukkadapu. 2010. Experiments on Indian Language Dependency Parsing. In Proc of *ICON-2010 tools contest on Indian language dependency parsing*. Kharagpur, India.

P. Mannem, A. Abhilash and A. Bharati. 2009. LTAGspinal Treebank and Parser for Hindi. *Proceedings of International Conference on NLP, Hyderabad. 2009.*

R. McDonald and J. Nivre. 2007. Characterizing the Errors of Data-Driven Dependency Parsing Models. In *Proc of Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*

J. Nivre. 2009. Non-Projective Dependency Parsing in Expected Linear Time. In *Proc. of ACL IJCNLP*.

J. Nivre and R. McDonald. 2008. Integrating graph-based and transition-based dependency parsers. In *Proc. of ACL-HLT.*

J. Nivre, J. Hall, S. Kubler, R. McDonald, J. Nilsson, S. Riedel and D. Yuret. 2007a. The CoNLL 2007 Shared Task on Dependency Parsing. In *Proc of EMNLP/CoNLL-2007.*

J. Nivre, J. Hall, J. Nilsson, A. Chanev, G. Eryigit, S. Kübler, S. Marinov and E Marsi. 2007b. MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2), 95-135.

D. Seddah, M. Candito and B. Crabbé. 2009. Cross parser evaluation : a French Treebanks study. *In Proceedings of the 11th IWPT09. Paris. 2009.*

R. Tsarfaty, D. Seddah, Y. Goldberg, S. Kuebler, Y. Versley, M. Candito, J. Foster, I. Rehbein and L. Tounsi. 2010. Statistical Parsing of Morphologically Ricj Languages (SPMRL) What, How and Wither. In Proc of *NAACL-HLT 2010 workshop on Statistical Parsing of Morphologically Rich Languages (SPMRL 2010), Los Angeles, CA.*

R. Tsarfaty and K. Sima'an. 2008. Relational-Realizational Parsing. *Proceedings of the 22nd CoLing. Manchester, UK.*

Team	*Average*	*Fine-grained*			*Coarse-grained*		
		Hindi	*Bangla*	*Telugu*	*Hindi*	*Bangla*	*Telugu*
Kosaraju et al.	76.83	88.63	70.55	70.12	88.87	73.67	69.12
Kolachina et al.	76.42	86.22	70.14	68.11	88.96	75.65	69.45
Attardi et al.	75.80	87.49	70.66	65.61	88.98	74.61	67.45
Abhilash and Mannem	72.59	82.18	65.97	66.94	83.41	69.09	67.95
Ghosh et al.	--		64.31				
Kesedi et al.	--						48.08

Table 4a: Labeled Attachment Score (LAS) of all the systems in the ICON 2010 Tools Contest.

Team	*Average*	*Fine-grained*			*Coarse-grained*		
		Hindi	*Bangla*	*Telugu*	*Hindi*	*Bangla*	*Telugu*
Attardi et al.	90.94	94.78	87.41	90.48	94.57	88.24	90.15
Kolachina et al.	90.46	93.25	87.10	90.15	94.13	88.14	89.98
Kosaraju et al.	90.30	94.54	86.16	91.82	93.62	86.16	89.48
Abhilash and Mannem	86.63	89.62	83.45	86.81	89.62	83.45	86.81
Ghosh et al.	--		83.87				
Kesedi et al.	--						76.29

Table 4b: Unlabeled Attachment Score (UAS) of all the systems in the ICON 2010 Tools Contest.

Team	*Average*	*Fine-grained*			*Coarse-grained*		
		Hindi	*Bangla*	*Telugu*	*Hindi*	*Bangla*	*Telugu*
Kosaraju et al.	79.14	90.00	73.36	71.95	90.79	77.42	71.29
Kolachina et al.	78.70	87.95	73.26	70.12	90.91	78.67	71.29
Attardi et al.	77.80	88.96	73.47	66.94	90.73	77.73	68.95
Abhilash and Mannem	75.37	84.02	69.51	69.62	85.29	73.15	70.62
Ghosh et al.	--		69.30				
Kesedi et al.	--						50.25

Table 4c: Label Score (LS) of all the systems in the ICON 2010 Tools Contest.

Appendix – I: Dependency labels

Tag Name	Tag description	Coarse-grain tag
k1	karta (doer/agent/subject)	k1
pk1	prayojaka karta (Causer)	k1
jk1	prayojya karta (causee)	vmod
mk1	madhyastha karta (mediator-causer)	vmod
k1g	gauna karta (secondary karta)	vmod
k1s	vidheya karta (karta samanadhikarana)	k1s
k2	karma (object/patient)	k2
k2p	Goal, Destination	k2p
k2g	gauna karma (secondary karma)	vmod
k2s	karma samanadhikarana (object complement)	k2s
k3	karana (instrument)	k3
k4	sampradaana (recipient)	k4
k4a	anubhava karta (Experiencer)	k4a
k5	apaadaana (source)	k5
k5prk	prakruti apadana ('source material' in verbs denoting change of state)	vmod
k7t	kaalaadhikarana (location in time)	k7
k7p	deshadhikarana (location in space)	k7
k7	vishayaadhikarana (location abstract)	k7
k*u	saadrishya (similarity)	vmod
k*s	samanadhikarana (complement)	vmod
r6	shashthi (possessive)	r6
r6-k1, r6-k2	*karta* or *karma* of a conjunct verb (complex predicate)	r6-k1, r6-k2
r6v	('kA' relation between a noun and a verb)	vmod
adv	kriyaavisheshana ('manner adverbs' only)	vmod
sent-adv	Sentential Adverbs	vmod
rd	prati (direction)	vmod
rh	hetu (cause-effect)	rh
rt	taadarthya (purpose)	rt

ras-k*	upapada__ sahakaarakatwa (associative)	vmod
ras-neg	Negation in Associatives	vmod
rs	relation samanadhikaran (noun elaboration)	rs
rsp	relation for duratives	nmod
rad	Address words	vmod
nmod__relc, jjmod__relc, rbmod__relc	Relative clauses, jo-vo constructions	relc
nmod__*inv		nmod
nmod	Noun modifier (including participles)	nmod
vmod	Verb modifier	vmod
jjmod	Modifiers of the adjectives	jjmod
rbmod	Modiiers of adverbs	rbmod
pof	Part of relation	pof
ccof	Conjunct of relation	ccof
fragof	Fragment of	fragof
enm	Enumerator	vmod
nmod__adj	adjectival modifications	nmod__adj
lwg__psp	noun and post-position/suffix modification	lwg__psp
lwg__neg	NEG and verb/noun modification	lwg__neg
lwg__vaux	Auxiliary verb modification	lwg__vaux
lwg__rp	particle modification	lwg__rp
lwg_cont	lwg continuation relation	lwg_cont
lwg__*	Other modifications in lwg	lwg__rest
jjmod__intf	intensifier adjectival modifications.	jjmod__intf
pof__redup	reduplication	pof__redup
pof__cn	compound noun	pof__cn
pof__cv	compound verb	pof__cv
rsym	Punctuations and symbols	rsym
mod	Modifier	mod