Exploring Semantic Information in Hindi WordNet for Hindi Dependency Parsing

Sambhav Jain Naman Jain Aniruddha Tammewar
Riyaz Ahmad Bhat Dipti Misra Sharma
Language Technologies Research Centre
IIIT Hyderabad
{sambhav.jain, riyaz.bhat}@research.iiit.ac.in, dipti@iiit.ac.in
{naman.jain, uttam.tammewar}@students.iiit.ac.in

Abstract

In this paper, we present our efforts towards incorporating external knowledge from Hindi WordNet to aid dependency parsing. We conduct parsing experiments on Hindi, an Indo-Aryan language, utilizing the information from concept ontologies available in Hindi WordNet to complement the morpho-syntactic information already available. The work is driven by the insight that concept ontologies capture a specific real world aspect of lexical items, which is quite distinct and unlikely to be deduced from morpho-syntactic information such as morph, POS-tag and chunk. This complementing information is encoded as an additional feature for data driven parsing and experiments are conducted. We perform experiments over datasets of different sizes. We achieve an improvement of 1.1% (LAS) when training on 1,000 sentences and 0.2% (LAS) on 13,371 sentences over the baseline. The improvements are statistically significant at $p<0.01$. The higher improvements on 1,000 sentences suggest that the semantic information could address the data sparsity problem.

1 Introduction

Last decade has witnessed several efforts towards developing robust data driven dependency parsing techniques (Kübler et al., 2009). The efforts, in turn, initiated a parallel drive for building dependency annotated treebanks (Tsarfaty et al., 2013), which serve as a data source for training data driven dependency parsers. The annotations are often multi-layered and furnish information on part of speech category of word forms, their morphological features, related word groups and the syntactic relations. The availability of such rich resources have considerably improved the parsing performance of syntactic parsers (Collins et al., 1999). However, the error analysis studies carried out on these parsers later revealed that certain syntactic relations are difficult to deduce and disambiguate with the syntactic information available in the annotated treebanks.

The need for richer information invoked several efforts in the direction of annotating higher order linguistic information in treebanks. It was felt that semantics can be leveraged for syntactic disambiguation and thus semantic annotation was performed in syntactic treebanks to complement the morpho-syntactic annotations (Kingsbury et al., 2002; Montemagni et al., 2003). Fujita et al. (2007) and MacKinlay et al. (2012) illustrated that semantic annotation delivers a significant improvement in parsing, confirming the hypothesis that semantics can assist syntactic analysis.

Among Indian languages, notable efforts on using semantic information in dependency parsing are on *Hindi*. Bharati et al. (2008) illustrated that mere *animacy* (human, non-human and inanimate) of a nominal significantly improves the accuracy of the parser. Later studies on extending such information with finer semantic distinctions like *time*, *place*, *abstract* reconfirmed the substantial role of semantics in syntactic parsing (Ambati et al., 2009). These studies are carried out on a dataset with hand annotated semantics. Although these studies provide deep insights on the role of semantics in parsing, they are limited in application as such information can not be automatically generated while parsing new sentences.

In this work, we make an effort to supply the aforementioned semantic information by employing concept hierarchy available in Hindi WordNet (henceforth HWN).

International Joint Conference on Natural Language Processing, pages 189–197,
Nagoya, Japan, 14-18 October 2013.

2 Related Work

Attempts have been made to utilize hand annotated semantic information for *constituency parsing* (Fujita et al., 2007; MacKinlay et al., 2012) as well as *dependency parsing* (Øvrelid and Nivre, 2007; Bharati et al., 2008; Ambati et al., 2009). However, acquiring such information for new sentences remains a challenge. This leads us to the exploration of lexical databases and ontologies for accessing semantic information useful for parsing. Xiong et al. (2005) used two lexical resources *HowNet*[1](Dong and Dong, 2000) and *TongYiCi CiLin* (Mei and Gao, 1996) for parsing Penn Chinese Treebank (Xue et al., 2002). Agirre et al. (2008) demonstrated that semantic classes obtained from English WordNet (Miller, 1995) help to obtain significant improvements in both PP attachment and PCFG parsing. Similarly, for dependency parsing, Agirre et al. (2011) utilized the English WordNet semantic classes and improved parsing accuracies.

3 Background and Challenges

Hindi is an Indo-Aryan language with richer morphology as compared to English. It exerts a relatively free word order with SOV being the default configuration. Due to the flexible word order, dependency representations are preferred over constituency for its syntactic analysis (Bharati and Sangal, 1993). The dependency representations do not constrain the order of words in a sentence and thus are better suited for flexible ordering of words. The dependency grammar formalism, used for Hindi is *Computational Paninian Framework* (CPG) (Begum et al., 2008; Bharati et al., 2009). The dependency relations in CPG formalism are closer to semantics and hence they are also denoted as *syntactico-semantic* relations.

The most important feature explored for dependency parsing is 'case clitics' that largely governs the relations nominals bear with their heads. Several efforts in past, on parsing Hindi, have greatly benefited by utilizing these clitics as a feature (Ambati et al., 2010a; Ambati et al., 2010b). However, case markers and case roles do not have a one-to-one mapping, each case marker is distributed over a number of case roles. Among the six case markers only Ergative case marker is unambiguous (Mohanan, 1994). Although case markers are good indicators of the relation a nominal bears in a sentence, their ambiguous nature bar their ability in effectively identifying the role of a nominal while parsing. Consider the examples from (1a-e), the instrumental se is extremely ambiguous. It can mark the instrumental adjuncts as in (1a), source expressions as in (1b), material as in (1c), comitatives as in (1d), and causes as in (1e).

(1a) मोहन ने चाबी से ताला खोला ।
Mohan-*Erg* key-*Inst* lock-*Nom* open
'Mohan opened the lock with a key.'

(1b) गीता ने दिल्ली से सामान मंगवाया ।
Geeta-*Erg* Delhi-*Inst* luggage-*Nom* procure
'Geeta procured the luggage from Delhi.'

(1c) मूर्तिकार ने पत्थर से मूर्ति बनायी ।
sculptor-*Erg* stone-*Inst* idol-*Nom* make
'The sculptor made an idol out of stone.'

(1d) राम की श्याम से बात हुई ।
Ram-*Gen* Shyaam-*Inst* talk-*Nom* happen
'Ram spoke to Shyaam.'

(1e) बारिश से कई फसलें तबाह
rain-*Inst* many crops-*Nom* destroy
हो गयीं ।
happen-*Perf*
'Many crops were destroyed due to the rain.'

Not all instances of a nominal in Hindi are case marked, as shown in Table 1. In appropriate contexts, a nominal can also bear a nominative case which is morphologically null (henceforth referred as unmarked nominals). It is possible, in fact quite frequent, to have more than one unmarked nominal within a single clause and due to the relative free word order, the movement can result in different surface configurations.

(2a) चिड़िया दाना चुग रही है ।
bird-*Nom* grain-*Nom* peck-*Prog*

(2b) दाना चिड़िया चुग रही है ।
grain-*Nom* bird-*Nom* peck-*Prog*
'A bird is pecking grain.'

	Patient-Unmarked	Patient-Marked
Agent-Unmarked	1276	741
Agent-Marked	5373	966

Table 1: Co-occurrence of Marked and Unmarked verb arguments in Hindi Dependency Treebank. *Source*: training-set, shared task MTPIL 2012

A conventional parser has no cues for the disambiguation of instrumental case marker se in examples (1a-e) and similarly, in example (2a-b), it

[1]http://www.keenage.com

is hard for the parser to know whether 'bird' or 'grain' is the agent of the action 'peck'. Apart from lexical and structural ambiguity, there are also data sparsity and out of vocabulary (OOV) problems when parsing out-of-domain text. Traditionally, syntactic parsing has largely been limited to the use of only a few lexical features. Features like POS-tags are way too coarse to provide deep information valuable for syntactic parsing. So in order to assist the parser for better judgments, we need to complement the morphology somehow.

4 Hindi WordNet and Concept Ontologies

Hindi WordNet is a lexical database developed on the lines of English Wordnet, under the Indo WordNet project (Narayan et al., 2002). For each lexical item, Hindi WordNet defines a synset which enlists its synonyms. Further, each synset is mapped to a concept ontology. The concept ontology is a hierarchical organization of concepts like entities, actions etc. which defines the semantic properties of lexical items of a given synset. The ontology consists of around 200 different concepts. The lexical item is the leaf node in this hierarchical construct. As we move up the hierarchy, the specific semantic aspects of a given lexical item are unraveled. The hierarchy terminates, immediately after capturing the syntactic category of a word, at the *TOP* node. The *TOP* acts as a $root$, holding the hierarchies of all the lexical items listed in HWN. Figure 1 illustrates a typical hierarchy in this ontology, where *Ape* is the most explanatory node. As we move up, it becomes more and more generic. Further, the relations between different synsets are captured based on the following paradigms :

- Semantic (hypernymy, hyponymy, meronymy etc.)
- Lexical (antonymy, synonymy etc.)
- Gradience (size, quality, manner etc.).

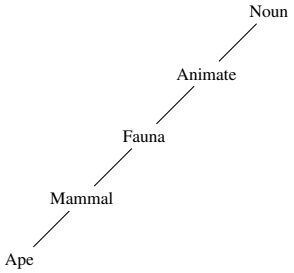


Figure 1: Sample Hierarchy of Concepts in Hindi Wordnet

Type	Sentence Count	Token Count	Chunk[4] Count
Training	12,038	268,009	142,445
Development	1,233	26,416	13,945
Testing	1,828	39,775	21,165

Table 2: Statistics of Data Sets used for experiments

5 Hindi Dependency Treebank

In this section, we give an overview of Hindi Treebank (HTB ver-0.51) (Bhatt et al., 2009; Palmer et al., 2009) a part of which was released for Hindi Dependency Parsing shared task, MTPIL, (Sharma et al., 2012). It is a multi-layered dependency treebank with morphological, part-of-speech and dependency annotations based on the Computational Paninian Framework (henceforth CPG). In the dependency annotation, relations are mainly verb-centric. The relation that holds between a verb and its arguments is called a '*karaka*' relation. Besides *karaka* relations, dependency relations also exist between nouns (genitives), between nouns and their modifiers (adjectival modification, relativization), between verbs and their modifiers (adverbial modification including subordination). CPG provides an essentially syntactico-semantic dependency annotation, incorporating *karaka* (e.g., agent, theme, etc.), *non-karaka* (e.g. possession, purpose) and other (part of) relations. A complete tag-set of dependency relations based on CPG can be found in (Bharati et al., 2009). The ones starting with '*k*' are largely Paninian *karaka* relations, and are assigned to the arguments of a verb. The data is released in two formats, SSF (Bharati et al., 2007) and CoNLL-X[2] formats (details in Table 2). It has also been released in UTF-8 encoding and roman readable WX[3] notation. We are using the CoNLL-X format and UTF-8 encoding.

6 Incorporating Knowledge from Concept Ontologies

In this section, we present our approach to incorporate semantic knowledge from HWN into the parsing model. We transform the hierarchical information in the concept ontology listed in HWN, into a string feature (henceforth WN feature) for

[2]http://ilk.uvt.nl/conll/#dataformat

[3]http://sanskrit.inria.fr/DATA/wx.html

[4]A chunk is a set of adjacent words which are in dependency relation with each other, and are connected to the rest of the words by a single incoming arc.

all the tokens in our data. Given a lexical item, we extract the information using its syntactic category from the ontological hierarchy corresponding to the most appropriate sense selected. In the following, we discuss in detail the selection and incorporation of this information with the challenges posed.

6.1 Feature Extraction

In this section, we explore the extraction of features from HWN corresponding to the lexical items in our data. We also address the issues like sense selection and coverage.

6.1.1 Sense Selection

Attributed to the phenomenon of lexical ambiguity, a lexical item can have senses varying across different contexts. Although HWN lists all the possible senses of a lexical item, to choose the contextually appropriate sense is a challenging task. Here, we discuss our approach to select the sense of a lexical item best suited in a given context.

- $Category\ Based\ Sense\ Selection$: Consider a word $chaat$, it can either mean 'lick' or 'snacks'. The former corresponds to a verb while the latter is a nominal as depicted in Figure 2. The syntactic category of a lexical item provides an initial cue for the sense selection. Among the varied senses, we filter out the senses that do not fall into its syntactic category.

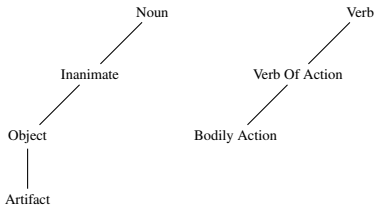


Figure 2: Nominal and Verb Sense of $chaat$

- $Intra-Category\ Sense\ Selection$: As a matter of fact, words are ambiguous not only across different syntactic categories but also within same category as depicted in Figure 3. Once the senses of a lexical item are filtered based on its syntactic category, within category senses, if many, are investigated for the best sense based on the following strategies:
  - $First\ Sense$: Among the varied senses, we select the first sense listed in HWN corresponding to the POS-tag of a given lexical item. The choice is motivated by our observation that the senses of a lexical item are ordered in the descending order of their frequencies of usage i.e., the first sense listed in HWN is the predominant sense of a given lexical item.
  - WSD: Although first sense captures the predominant usage of a lexical item, it is inappropriate for its other infrequent usages. We, therefore, need to pick the contextually appropriate sense of a lexical item. To this end, we exercise Extended Lesk, a classical word sense disambiguation algorithm (Banerjee and Pedersen, 2003).

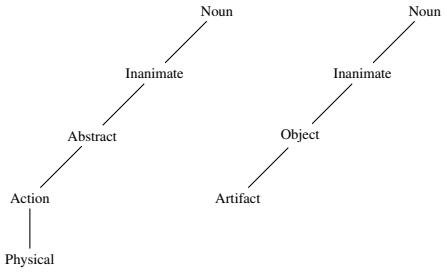


Figure 3: Two senses for the nominal $chaat$

6.1.2 Numeric Expressions

As is obvious, no lexical resource can have an exhaustive coverage because of the evolving nature of human language. In the context of HWN, the problem further intensifies as it restricts the entry to only words of open class syntactic categories. Apart from that, it also has a limited coverage for numeric expressions as these expressions belong to an infinite set. Numerals can be used in wide range of senses. Apart from their simple ordinal or cardinal usages, they can also be used as nominals in expressions like time and measurement. In their adjectival sense, WN features can be extracted corresponding to the head word they modify e.g., the temporal sense of an expression 10 $saal$ can be identified by the head word $saal$ 'year'. However, to identify the temporal sense of a numeral, used as nominal, like 2013 is challenging. We use a numeric-expression recognizer, built in-house, to identify measurement and temporal expressions. The tool makes use of regular expressions and cue words. Once identified, we assign them an appropriate HWN ontological hierarchy which either corresponds to *time*, *measurement* or *number*.

6.1.3 Complex Predicate as a Feature

Complex predicates (CPs, also known as complex verbs) are highly frequent in South Asian languages (Mohanan, 1997). They occur in the form of nominal+verb combinations (called conjunct verbs) and verb+verb combinations (called compound verbs). For example, in (5), 'शरण लेना' (refuge take) is a complex predicate composed of a nominal 'शरण' and a light verb 'लेना'. The constituents of a complex predicate are related by a dependency relation *pof* in HDT. In Hindi dependency parsing, the major chunk of parse errors is attributed to the low learnability of complex predicates (Husain and Agrawal, 2012). Begum et al. (2011) addressed the identification of these expressions using some linguistic rules. Fortunately, HWN has listed a finite set of these expressions in its database (Chakrabarti et al., 2007). We first extract the multi word expressions listed in HWN if the last word in the expression is a verb. Then from the list only 2-word expressions are selected and treated as complex predicates. Instead of adding WN features to the nominal of a complex predicate, we assign a separate CP tag to it. The semantics of light verbs is, however, kept as such.

6.2 Feature Design

After the extraction of WN features, we explore possibilities of their design and incorporation in the parsing framework, as follows.

6.2.1 Grouping Similar Features

We observed that few concept ontological lineages are semantically similar. For example, the six lineages depicted below address the notion of time.

- *Time*
- *Descriptive→Time*
- *Inanimate→Abstract→Time*
- *Inanimate→Abstract→Time→Period*
- *Inanimate→Abstract→Time→Season*
- *Inanimate→Abstract→Time→Mythological Period*

Since our focus is on adding representative semantic features which can assist parsing, we believe that such divergences should be grouped together. In the listed example, first, second and the last four differ in terms of their origin and belong to different branches in the hierarchy. Thus they can not be grouped by optimal depth selection (described later in Section 6.2.3) and requires a manual scrutiny. We studied the possible lineages in the concept ontology and performed merging wherever necessary, furnishing a semantically well diverse set of concept lineages.

6.2.2 Split Vs Conjoined

The concept lineage, derived for a word from HWN concept ontology, contains diverse concepts at each level of the lineage. The choice of using each of these concepts as independent features or the complete lineage as a single feature demands exploration. In the context of parsing, each independent concept from the lineage can potentially capture a specific aspect of syntax, depending on the fineness of the concept. The down side of this proposition is the increase in the feature dimensions, as each level adds a new dimension in the feature space. Whereas, using the complete lineage as a single feature does not add any additional dimension in the feature space but captures only a specific concept. This trade off is difficult to comprehend on theoretical grounds, hence we explore both choices of feature design in our experiments.

6.2.3 Ontology Depth

Hindi WordNet concept ontology furnishes a 'generalization hierarchy' for a lexical item, where the specificity of concepts increases as we move down the hierarchy. It may look intuitive to use fully expanded concept lineage, as it contains more detailed description of the lexical unit. However, opting for a highly fine-grained concept lineage leads to the problem of sparseness. It becomes less and less probable to find ample training examples as the feature becomes more fine-grained. At the same time, too much generalization is also unrewarding since the richer information is cast away in the excessive coarser lineage. This calls for measures to obtain an optimal depth of concept lineage for each lexical item. On one hand it should be generalized enough to give significant examples of its respective type while on the other hand, it should be fine enough to capture the rich ontological concept associated with the lexical unit. In order to quantify the trade-off we resort to statistical correlation measures and employed *Gini Coefficient* (Gini, 1912). We computed the coefficient against all possible concept lineages in the training set and set a threshold. The lineages that fall below the threshold are generalized till they are above the threshold. For example, in Figure 1 the concept ape is suppressed to give the lineage till $mammal$ only. So in future if a word gives the lineage as in Figure 1 it will be

replaced with its one level up generalization i.e. *Animate→Fauna→Mammal*.

7 Experiments and Results

In our experiments, we focus on establishing dependency relations between the chunk heads which we henceforth denote as *inter-chunk* parsing. The relations between the tokens of a chunk (*intra-chunk* dependencies) are not considered for experimentation. In example (3), dotted line shows an *intra-chunk* relation while the bold lines show *inter-chunk* dependency relations[5]. The decision is motivated by the fact that the *intra-chunk* dependencies can easily be predicated automatically using a finite set of rules (Kosaraju et al., 2012). Moreover we also observed the high learnability of *intra-chunk* relations from an initial experiment. We found the accuracies of *intra-chunk* dependencies to be more than 99.00% for both Labeled Attachment and Unlabeled Attachment.

In this section, we present our parsing experiments incorporating the features extracted from HWN, as discussed in Section 6. First we setup our baseline parser followed by the detailed discussion on the impact of the individual features, extracted from HWN, on the overall parsing performance.

We setup our baseline parser on the lines of (Singla et al., 2012) with minor modifications in the parser *feature model*. We employ MaltParser version-1.7[6] (Nivre et al., 2007) and Nivre's Arc Eager algorithm for all our experiments reported in this work. All the results reported are evaluated using *eval07.pl*[7]. We use MTPIL (Sharma et al., 2012) dependency parsing shared task data described in Section 5. Among the features available in the FEATS column of the CoNLL format data, we only consider *Tense, Aspect, Modality (tam)* and *postpositions* while training the baseline parser. Other columns like POS, LEMMA, etc. are used as such. After the baseline, the parsing framework is further enriched with the semantic features extracted from HWN to address the problems raised in Section 3. These features are added in the FEATS column of the data, separated by '|'. In a pilot experiment split form of features, as discussed in Section 6.2.2, are found to perform better than conjoined form, which motivate us to use WN feature in split form in all our experiments. The experimentation proceeds in the order as listed in Table 3 which also presents the consolidated results of our parsing experiments using the MTPIL training and testing sets. In order to see the impact of semantic information on data sparsity, we split the MTPIL training set into datasets of different sizes. We experiment with 6 data sets of different sizes. The results are produced on MTPIL test set and are plotted on Graph (Figure 4). The increase in LS and LAS, as the training size decreases, shows the impact of semantic information on data sparsity. The improvement of 1.1 (LAS) by semantics upon reducing the training examples to 1000 implies that semantics can address the data sparsity and OOV problems when working with out-of-domain text.

Next we discuss the impact of WN features on the accuracy of our parsing results produced on datasets of different sizes:

- *Sense Selection*: As discussed in Section 6.1.1, we perform two experiments to extract the WN features corresponding to the most appropriate sense of a lexical item. In the first experiment, the first sense of each lexical item is selected while in the second, WSD is used to pick the contextually most appropriate sense. These features corresponding to the chosen sense are coupled with the features already present in the baseline. As depicted in Graph (Figure 4), there is a average increase of 0.38 (LAS) on all datasets using the first sense strategy from the baseline. However, using WSD the accuracy decreased across all datasets. As is obvious, the fall in accuracy can be attributed to the wrong sense selection. The problem can be addressed by using better WSD algorithms for Hindi.

- *Numeric Expressions and Grouping*: As discussed in Section 6.1.2, numeric expressions and sense grouping increases the coverage of HWN. This obvious reason is clearly depicted in the improvement in parsing results as shown in Table 3. More the semantic information available in the data, more will be its impact on the parsing.

- *Depth of Information*: The optimality of feature coarseness is put to test in this ex-

[5] k1: Doer, k1s: Noun Complement, k5: Source, k7p: Place, k7t: Time, pof: part-of (complex predicate), lwg__psp: local-word-group postposition

[6] http://www.maltparser.org/download.html

[7] http://nextens.uvt.nl/depparse-wiki/SoftwarePage/#eval07.pl

periment. This experiment is run on numeric expression data with feature pruning done as described in Section 6.2.3. An increment of average 0.03% LAS across datasets is observed from the previous experiment. In the test set, there are only a few cases that are updated by choosing an optimal lineage depth which explains the minimal increase in accuracy.

- *Complex Predicate*: As pointed in (Begum et al., 2011), addressing the low learnability of complex predicates can improve the parsing results. The improvements are particularly seen in the core arguments of a verb. The similar syntactic distribution of adjectival or nominal element of a complex predicate and the syntactic arguments of a verb particularly *objects*, make these expressions highly ambiguous. Identifying these expressions beforehand, as suggested in (Begum et al., 2011), improves the parsing performance. The incorporation of this crucial information from HWN is rewarding as we achieve an improvement of ∼0.4% in LAS on a dataset of 1,000 sentences.

	Experiments	LAS(%)	UAS(%)	LS(%)
E1	Baseline	83.69	92.43	86.58
E2	E1 + First Sense	83.78	92.4	86.73
E3	E1 + WSD (Extended Lesk)	83.6	92.34	86.57
E4	E2 + Numeric Expressions & Grouping	**83.88**	**92.45**	**86.87**
E5	E4 + Ontological Depth	83.84	92.4	86.79
E6	E4 + Complex Predicate	83.75	92.39	86.72
E7	E5 + E6 (Complex Predicate + Ontological Depth)	83.74	92.39	85.7

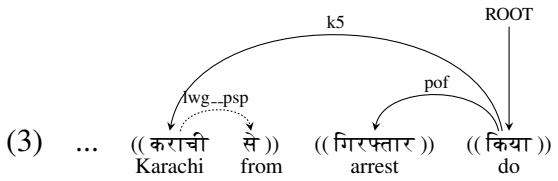
Table 3: Results of Parsing Experiments

8 Discussion

In this section, we discuss further, how well the issues raised in Section 3 are handled by the incorporation of semantic information in the parsing framework of Hindi. In Section 3, we stated that ambiguities in morphological cases in Hindi bar their efficient exploitation while parsing. Also we noted that unmarked nominals may as well affect the performance of a parser. So we propose semantics as a complementing information that can fill these gaps. Below we discuss whether semantic information has bridged these gaps or not.

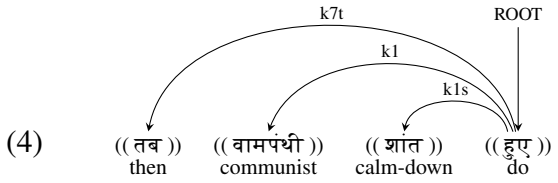
- *Case Ambiguity*: Including the semantics from HWN to help disambiguate the confusion present in a case marker, has improved parsing accuracy. Particularly confusion among the roles of concrete vs abstract time and place, and direct vs indirect object relations has been removed. In example (3), the dependency relation between nodes *Karachi* and *do* has been corrected from $k2$ 'Theme' to $k5$ 'Source'. The postposition *from* can either mark a theme or a source relation. Semantics has removed this confusion.

(3) ...



- *Lack of Case Marker*: In absence of case marking lexical semantics acted as a complementing information. The improvement has been, as observed during error analysis, particularly for agents and patients. Thus semantics can be seen here as pseudo case markers. This is clearly visible from the example (4). The dependency relation between the nodes *then* and *do* has been corrected to $k7t$ 'time of action' from $k1$ 'subject'.

(4)



- *Complex Predicates*: As we discussed, complex predicates are identified using HWN, so that the similar syntactic distributions of verb arguments and the nominal or adjectival part of a CP can be disambiguated. Identifying the complex predicates has turned to be rewarding. As was expected, the prior identification of CPs has significantly improved the joint identification of label and attachment. The system trained on 1,000 sentences has shown an improvement of 0.34% (LAS) and 0.2% (UAS) by prior identification of complex predicates. The confusion that has been removed is among the arguments of a verb and the nominal part of the CP i.e., between agent, patient vs nominal,

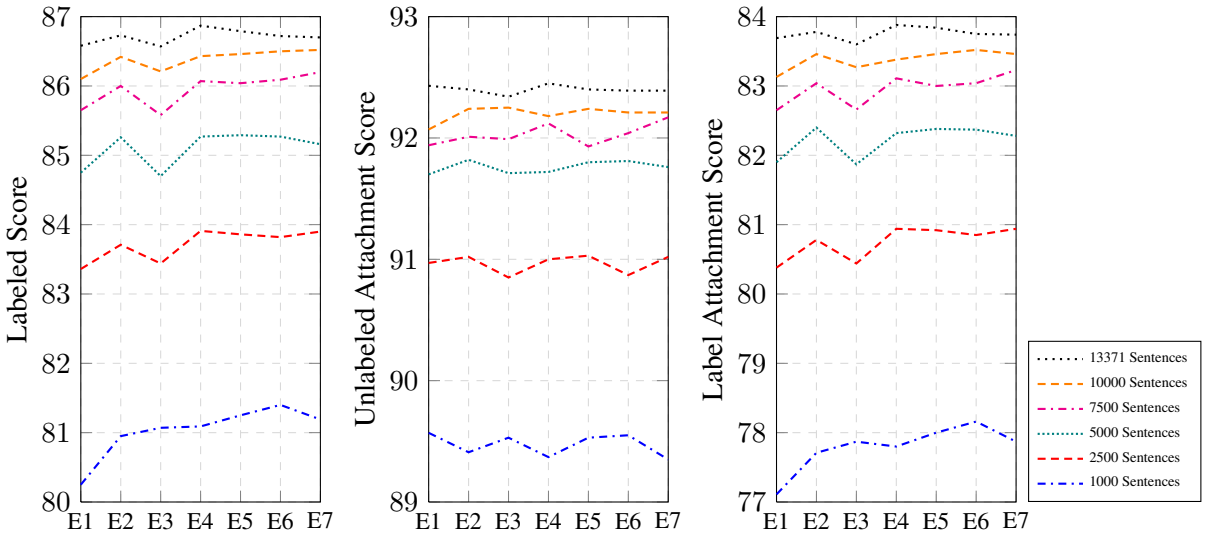


Figure 4: Impact of WN Features on Different Data Sizes

adjectival part of CP. In example below, baseline incorrectly identifies $refuge$ as an argument of verb $take$. '$refuge\ take$' is a complex predicate which is correctly identified upon incorporation of complex predicates in our parsing module.

(5) ... ((द्वीप island में in)) ((शरण refugee)) ((लिए take)) ...
k7p
lwg__psp
pof

9 Conclusion and Future Work

We present our efforts on exploring lexical resources, Hindi WordNet in our case, to discover features which complement the available morpho-syntactic feature conventionally explored for parsing. We find concept ontology available in HWN quite resourceful in furnishing features which can essentially break syntactic ambiguity, resulting in better accuracies for parsing. In future we would like to investigate other hierarchies like hypernymy, hyponymy, meronymy etc. We would also like to substitute lexical units with their respective synsets as proposed in (Agirre et al., 2011).

References

E. Agirre, T. Baldwin, and D. Martinez. 2008. Improving parsing and PP attachment performance with sense information. *Proceedings of ACL-08: HLT*, pages 317–325.

E. Agirre, K. Bengoetxea, K. Gojenola, and J. Nivre. 2011. Improving dependency parsing with semantic classes. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 699–703.

Bharat Ram Ambati, Pujitha Gade, Chaitanya Gsk, and Samar Husain. 2009. Effect of minimal semantics on dependency parsing. In *Proceedings of the Student Research Workshop*, pages 1–5, Borovets, Bulgaria, September. Association for Computational Linguistics.

Bharat Ram Ambati, Samar Husain, Sambhav Jain, Dipti Misra Sharma, and Rajeev Sangal. 2010a. Two methods to incorporate local morphosyntactic features in Hindi dependency parsing. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 22–30. Association for Computational Linguistics.

B.R. Ambati, S. Husain, J. Nivre, and R. Sangal. 2010b. On the role of morphosyntactic features in Hindi dependency parsing. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 94–102. Association for Computational Linguistics.

Satanjeev Banerjee and Ted Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. In *International Joint Conference on Artificial Intelligence*, volume 18, pages 805–810. LAWRENCE ERLBAUM ASSOCIATES LTD.

R. Begum, S. Husain, A. Dhwaj, D.M. Sharma, L. Bai, and R. Sangal. 2008. Dependency annotation scheme for Indian languages. In *Proceedings of IJCNLP*. Citeseer.

Rafiya Begum, Karan Jindal, Ashish Jain, Samar Husain, and Dipti Misra Sharma. 2011. Identification of conjunct verbs in hindi and its effect on parsing accuracy. In *Computational Linguistics and Intelligent Text Processing*, pages 29–40. Springer.

A. Bharati and R. Sangal. 1993. Parsing free word order languages in the Paninian framework. In *Proceedings of the 31st annual meeting on Association for Computational Linguistics*, pages 105–111. Association for Computational Linguistics.

A. Bharati, R. Sangal, and D.M. Sharma. 2007. Ssf: Shakti standard format guide. *Language Technologies Research Centre, International Institute of Information Technology, Hyderabad, India*, pages 1–25.

A. Bharati, S. Husain, B. Ambati, S. Jain, D. Sharma, and R. Sangal. 2008. Two semantic features make all the difference in parsing accuracy. *Proc. of ICON*, 8.

A. Bharati, D.M. Sharma, S. Husain, L. Bai, R. Begum, and R. Sangal. 2009. AnnCorra: TreeBanks for Indian Languages Guidelines for Annotating Hindi TreeBank (version–2.0).

R. Bhatt, B. Narasimhan, M. Palmer, O. Rambow, D.M. Sharma, and F. Xia. 2009. A multi-representational and multi-layered treebank for hindi/urdu. In *Proceedings of the Third Linguistic Annotation Workshop*, pages 186–189. Association for Computational Linguistics.

Debasri Chakrabarti, Vaijayanthi Sarma, and Pushpak Bhattacharyya. 2007. Complex predicates in indian language wordnets. *Lexical Resources and Evaluation Journal*, 40(3-4).

Michael Collins, Lance Ramshaw, Jan Hajič, and Christoph Tillmann. 1999. A statistical parser for czech. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 505–512. Association for Computational Linguistics.

Zhendong Dong and Qiang Dong. 2000. Hownet chinese-english conceptual database. Technical report, Technical Report Online Software Database, Released at ACL. http://www. keenage. com.

Sanae Fujita, Francis Bond, Stephan Oepen, and Takaaki Tanaka. 2007. Exploiting Semantic Information for HPSG Parse Selection. In *ACL 2007 Workshop on Deep Linguistic Processing*, pages 25–32, Prague, Czech Republic, June. Association for Computational Linguistics.

Corrado Gini. 1912. *Variabilità e mutabilità: contributo allo studio delle distribuzioni e delle relazioni statistiche.* Tipogr. di P. Cuppini.

Samar Husain and Bhasha Agrawal. 2012. Analyzing Parser Errors to improve parsing accuracy and to inform tree banking decisions. *Linguistic Issues in Language Technology*, 7(1).

Paul Kingsbury, Martha Palmer, and Mitch Marcus. 2002. Adding semantic annotation to the penn treebank. In *Proceedings of the Human Language Technology Conference*, pages 252–256. Citeseer.

Prudhvi Kosaraju, Samar Husain, Bharat Ram Ambati, Dipti Misra Sharma, and Rajeev Sangal. 2012. Intra-chunk dependency annotation: expanding Hindi inter-chunk annotated treebank. In *Proceedings of the Sixth Linguistic Annotation Workshop*, pages 49–56. Association for Computational Linguistics.

Sandra Kübler, Ryan McDonald, and Joakim Nivre. 2009. Dependency parsing. *Synthesis Lectures on Human Language Technologies*, 1(1):1–127.

Andrew MacKinlay, Rebecca Dridan, Diana McCarthy, and Timothy Baldwin. 2012. The effects of semantic annotations on precision parse ranking. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, pages 228–236. Association for Computational Linguistics.

Jia-ju Mei and Yunqi Gao. 1996. Tongyi cilin (a chinese thesaurus). *China: Shanghai Lexicographical Publishing House*.

George A Miller. 1995. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41.

Tara Mohanan. 1994. *Argument structure in Hindi*. Stanford Univ Center for the Study.

Tara Mohanan. 1997. Multidimensionality of representation: Nv complex predicates in hindi. *Complex predicates*, pages 431–471.

Simonetta Montemagni, Francesco Barsotti, Marco Battista, Nicoletta Calzolari, Ornella Corazzari, Alessandro Lenci, Antonio Zampolli, Francesca Fanciulli, Maria Massetani, Remo Raffaelli, et al. 2003. Building the Italian syntactic-semantic treebank. In *Treebanks*, pages 189–210. Springer.

Dipak Narayan, Debasri Chakrabarty, Prabhakar Pande, and Pushpak Bhattacharyya. 2002. An experience in building the indo wordnet-a wordnet for hindi. In *First International Conference on Global WordNet, Mysore, India*.

Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülsen Eryigit, Sandra Kubler, Svetoslav Marinov, and Erwin Marsi. 2007. MaltParser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95.

Lilja Øvrelid and Joakim Nivre. 2007. When word order and part-of-speech tags are not enough–Swedish dependency parsing with rich linguistic features. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP)*, pages 447–451.

M. Palmer, R. Bhatt, B. Narasimhan, O. Rambow, D.M. Sharma, and F. Xia. 2009. Hindi Syntax: Annotating Dependency, Lexical Predicate-Argument Structure, and Phrase Structure. In *The 7th International Conference on Natural Language Processing*, pages 14–17.

Dipti Misra Sharma, Prashanth Mannem, Joseph vanGenabith, Sobha Lalitha Devi, Radhika Mamidi, and Ranjani Parthasarathi, editors. 2012. *Proceedings of the Workshop on Machine Translation and Parsing in Indian Languages*. The COLING 2012 Organizing Committee, Mumbai, India, December.

Karan Singla, Aniruddha Tammewar, Naman Jain, and Sambhav Jain. 2012. Two-stage Approach for Hindi Dependency Parsing Using MaltParser. *Training*, 12041(268,093):22–27.

Reut Tsarfaty, Djamé Seddah, Sandra Kübler, and Joakim Nivre. 2013. Parsing Morphologically Rich Languages: Introduction to the Special Issue. *Computational Linguistics*, 39(1):15–22.

D. Xiong, S. Li, Q. Liu, S. Lin, and Y. Qian. 2005. Parsing the penn chinese treebank with semantic knowledge. *Natural Language Processing–IJCNLP 2005*, pages 70–81.

Nianwen Xue, Fu-Dong Chiou, and Martha Palmer. 2002. Building a large-scale annotated Chinese corpus. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–8. Association for Computational Linguistics.