

Issues in Analyzing Telugu Sentences towards Building a Telugu Treebank

Chaitanya Vempaty, Viswanatha Naidu, Samar Husain, Ravi Kiran, Lakshmi Bai,
Dipti M Sharma, and Rajeev Sangal

Language Technologies Research Centre, IIIT-Hyderabad, India
srpchaitanya@students.iiit.ac.in
{vnaidu,samar,ravikiranv}@research.iiit.ac.in
{lakshmi,dipti,sangal}@mail.iiit.ac.in

Abstract. This paper describes an effort towards building a Telugu Dependency Treebank. We discuss the basic framework and issues we encountered while annotating. 1487 sentences have been annotated in Paninian framework. We also discuss how some of the annotation decisions would effect the development of a parser for Telugu.

1 Introduction

Currently, an effort is underway to develop a large scale treebank for Indian Languages (ILs). Lack of such resources has been a major limiting factor in the development of good natural language processing tools. It is well known that the use of Phrase Structure (PS), is not well-suited for free word order languages [16]. Instead, the dependency framework appears to be better suited [11], [13], [6]. The effort described in this paper follows the Paninian grammatical framework [6] which is a dependency based approach. Recently, the Paninian framework has been successfully used for Hindi¹ dependency annotation [2]. This paper introduces how this framework can be used for analyzing Telugu. Telugu², an IL, is a language with relatively high free word-order. It is also morphologically very rich.

In the past, there has been significant amount of work in preparing such annotated linguistic resources, most notably the Penn treebank (PTB) [14] for English and Prague Dependency treebank (PDT) [10] for Czech. PTB uses the Phrase Structure annotation scheme whereas PDT implements a three layered annotation scheme, namely morphological, analytical (shallow dependency syntax) and tectogrammatical (deep dependency syntax). Other major efforts in the dependency framework are Alpino [17] for Dutch, [15] for English, TUT [9] for Italian, TIGER [8] for German, a multi-representational and multi-layered treebank for Hindi/Urdu [7]. Development of a Latin Dependency Treebank (LDT) for Latin is also an ongoing work [1].

In our treebank each sentence was manually pos-tagged and chunked³. They were then annotated for dependency relations. While chunking, we assumed that a chunk

¹ Hindi is South Asian Language and an official language of India spoken by 300 million people.

² Telugu is a Dravidian language and an official language of India spoken by 75 million people.

³ Akshar Bharati, Rajeev Sangal, Dipti Misra Sharma and Lakshmi Bai. 2006. AnnCorra: Annotating Corpora Guidelines for POS and Chunk Annotation for Indian Languages. Technical Report (TR-LTRC-31), Language Technologies Research Centre IIIT-Hyderabad. <http://ltrc.iiit.ac.in/MachineTrans/publications/technicalReports/tr031/posguidelines.pdf>

is a minimal, non-recursive structure consisting of correlated groups of words. Karaka relations (discussed in section 2) were marked between chunk heads, as the emphasis was on showing the right modifier-modified relationship and to ignore local details⁴.

For the following sentence all possible word combinations are grammatical.

((rAmudu_NNP))__NP ((paMdu_NN))__NP ((wiMtAdu_VM))__VGF.

rAmudu paMdu wiMtAdu.
 'Ram' 'fruit' 'eats'.
 Ram eats a fruit.

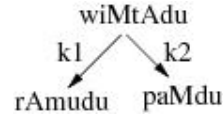


Fig. 1. Dependency Tree

In this paper we specifically discuss in detail, the following linguistic constructions::

- 1) Genitives:: In Telugu the genitive marker is often dropped.
- 2) Conjuncts:: Different constructions where a conjunct presence is explicit/implicit.
- 3) Copula :: Missing verbs i.e verbs are dropped.
- 4) Ani⁵ constructions:: Various ways of using the lexical item "ani" in language.

All the sentences are in 'wx' notation⁶ and their annotation is done in SSF (Shakti Standard Format)⁷.

2 Overview of Annotation Scheme

As mentioned earlier the annotation is done based on the Paninian grammatical framework that has been successfully used for developing Hyderabad dependency treebank [2] (HyDT). The annotation scheme considers the verb as the central, binding element of the sentence. In other words, the verb's requirements for its arguments is the starting point of the analysis. The relationship between the participant and the activity/state denoted by the verb is marked using relations that are called *karaka*.

It has been shown that the notion of karaka incorporates the local semantics of a verb in a sentence and that it is syntactico-semantic [6], [18]. For example, *karta* or *k1* is a relation that describes an argument that is most central to the action described by the verb. There are 6 basic karakas, namely; *adhikarana* 'location(k7)', *apadaan* 'source(k5)', *sampradaan* 'recipient(k4)', *karana* 'instrument(k3)', *karma* 'theme(k2)',

⁴ Intra-chunk dependencies are easy to mark and a rule-based system can be developed with high performance in automatically marking the intra-chunk relations. Due to the lack of space we do not elaborate it here.

⁵ Quotative marker in Telugu.

⁶ In this notation, capitalization roughly means aspiration for consonants and longer length for vowels. In addition, 'w' represents 't' as in French *entre* and 'x' means something similar to 'd' in French *de*, hence the name of the notation. http://ltrc.iiit.net/anusaaraka/SAN_MO/help.html#sec-b

⁷ SSF: Shakti Standard Format Guide. Akshar Bharati, Rajeev Sangal and Dipti Misra Sharma. Technical Report no: IIIT/TR/2009/85. http://www.iiit.ac.in/techreports/2009_85.pdf

karta ‘agent(k1)’. Other than the basic karaka relations the scheme has other relations such as ‘nmod’, ‘vmod’, ‘r6’ etc. The scheme has around 28 tags⁸. The tags are hierarchical. Figure 2 shows the hierarchy of the tagset.

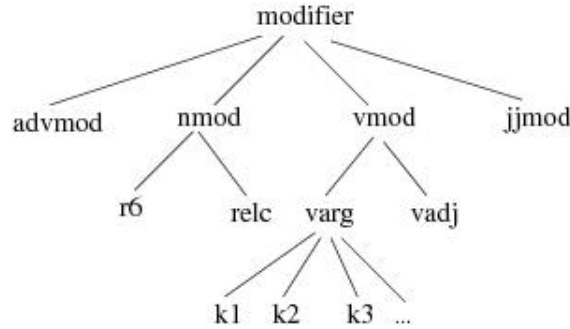


Fig. 2. Heirarchichy of tags

‘Advmod’, ‘nmod’, ‘vmod’ and ‘jjmod’ correspond to the adverb modifier, noun modifier, verb modifier and adjective modifier respectively. Below the noun modifier, we have the noun dependencies of r6 (possession) and relc (relative clause). Similarly, below the verb modifier we have the verb arguments, which are the karaka labels, k1, k2, k3 and so on.

3 Dependency Relations

The relations in the scheme are marked between chunk heads. The verb in simple sentence generally becomes the head of the sentence. The arguments of the verb are shown with appropriate labels. Figure 3(a),(b),(c) shows this for verbs ‘velwAdu’, ‘koVswAdu’, ‘iccAdu’ respectively. Likewise, noun becomes the head in the case of genitives etc. This can be seen in Figure 3(d).

- a. rAmudu hyderabad ki velwAdu.
‘Ram’ ‘hyderabad’ ‘to’ ‘go_will’.
Ram will go to hyderabad.
- b. rAmudu cAku wo paMdu koVswAdu.
‘Ram’ ‘knife’ ‘with’ ‘fruit’ ‘cuts’.
Ram cuts the fruit with knife.
- c. rAmudu sIwa ki apple iccAdu.
‘Ram’ ‘Sita’ ‘to’ ‘apple’ ‘gave’.
Ram gave an apple to Sita.
- d. rAmudi yoVkka puswakaM.
‘Ram’ ‘s’ ‘book’.
Ram’s book.

⁸ See <http://ltrc.iiit.ac.in/MachineTrans/research/tb/dep-tagset.pdf>

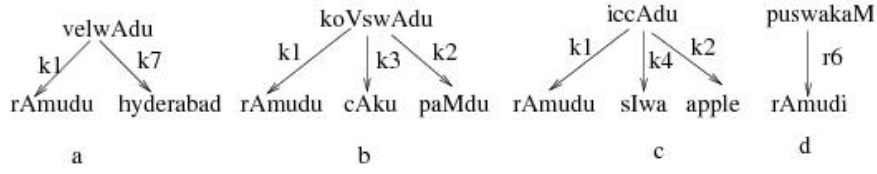


Fig. 3. Dependency Relations

The relation used for relative clauses is `nmod_relc`. Conjunct relations are treated as distinct from normal dependency relations. In this scheme [2] the conjuncts become the head. This is true for both coordinating & subordinating conjuncts.

4 Issues

In this section, we describe some issues we encountered while annotating the sentences and show how we analyzed them.

4.1 Genitives

Genitive is the case that marks a noun as modifying another noun. The relation between the Genitive noun and its head noun is denoted by "r6". In Telugu this can be exhibited broadly in two ways::

- Using explicit Genitive marker "yoVkka"

rAmudi yoVkka puswakaM.
 ‘Ram’ ‘s’ ‘book’.
 Ram’s book.

- Genitive marker is dropped::

In Telugu generally the masculine nouns⁹ have a possessive marker "i" indicating an implicit genitive marker. And in case of feminine nouns¹⁰ however this relationship must be inferred.

rAmudi puswakaM	sIwa puswakaM
‘Ram’ ‘-s’ ‘book’	‘Sita’ ‘book’
Ram’s book	Sita’s book

Yet, some masculine nouns (where the the lexical item and it’s root are identical) also exhibit this property.

raGu puswakaM.
 ‘raGu’ ‘book’.
 Raghu(’s) book.

⁹ Ram is a masculine noun.

¹⁰ Sita is a feminine noun.

Decision: If the genitive marker is not present, the manual annotator will have to infer the relation based on the context. Initial inter-annotator agreement is high which suggests that native Telugu speakers can easily identify this relation based on the context.

4.2 Conjuncts

In Telugu, conjuncts can occur as suffixes, lexical items and as DheerGaas¹¹

- Suffixes:: Conjuncts occur as TAM¹² of the verb.

nenu iMtiki velwe nidrapowAnu.
 'I' 'house_to' 'go_if' 'sleep_will'.
 I will sleep if I go home.

Decision: They are treated as *vmod of the type subordinating conjuncts* as shown in Figure 4.

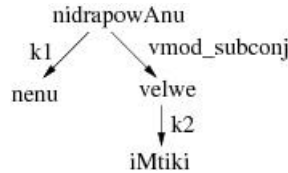


Fig. 4. Suffixes

- Lexical items:: They occur as *mariyu (and)*, *kAni (but)* etc...

rAmudu iMtiki vellAdu mariyu mohana mArket ki vellAdu.
 'Ram' 'house_to' 'went' 'and' 'mohan' 'market_to' 'went'.
 Ram went home and Mohan went to the market.

Decision: Handling simple coordinating conjuncts is straight forward. Figure 5 shows that their analysis in Telugu is consistent with the Hindi annotation.

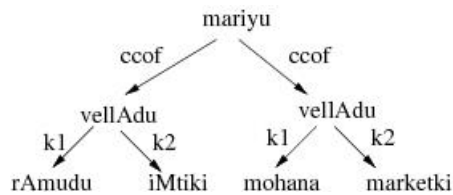


Fig. 5. Lexical Items

¹¹ DheerGaas are elongated forms of the vowels at the end of the lexical items [12].

¹² TAM - Tense Aspect Modality.

- DheerGaas:: By elongating the vowel at the end of the lexical items, the information of conjunction is implicit.

rAmudU sIwa iMtiki vellAru.
 ‘Ram’ ‘-and’ ‘Sita’ ‘home_to’ ‘went’.
 Ram and Sita went home.

Decision: A null element with a special tag, NULL_CCP¹³, is introduced.

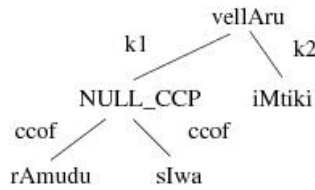


Fig. 6. dheergas

4.3 Copula

Copula is a linking verb. It is generally dropped in Telugu, unlike Hindi and English in which it takes the form "hE" and "be" respectively.

rAmudu maMci bAludu.
 ‘Ram’ ‘good’ ‘boy’.
 rAma accA laDakA hE¹⁴.
 Ram is a good boy.

Decision: An element with tag NULL_VG is introduced inorder to fit the criteria of the dependency schema which states that the root of a dependency tree¹⁵ is a main verb.

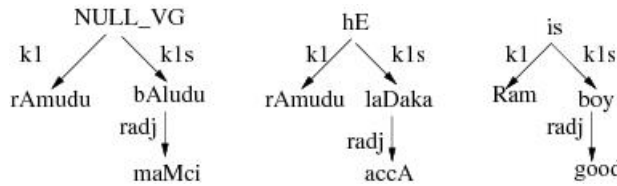


Fig. 7. Copula

The following are some more examples in which a verb is missing. NULL_VG inserted sentences are also given.

¹³ When one or more element from a sentence is dropped, it is called ellipses. A null element marked with a special tag ‘NULL’ is introduced in such cases. Note that without inserting a NULL the tree cannot be drawn. NULL_NP, NULL_VG, NULL_CCP etc mark different kinds of ellipses.

¹⁴ The sentence is a Hindi sentence.

¹⁵ k1s stands for k1 samanadhikaran which means of equal status to k1.

1. rAmudu bAludu mariyu allari pillavAdu.
 ‘Ram’ ‘boy’ ‘and’ ‘mischievous’ ‘kid’.
 Ram is a boy and a mischievous kid.
 rAmudu bAludu mariyu allari pillavAdu NULL_VG.
2. rAmudu maMci bAludu mariyu pallu wiMtAdu.
 ‘Ram’ ‘good’ ‘boy’ ‘and’ ‘fruits’ ‘eats’.
 Ram is a good boy and eats fruits.
 rAmudu maMci bAludu NULL_VG mariyu paMdu wiMtAdu.

4.4 “ani” Constructions

There are broadly two different senses for the lexical item ‘ani’.

- a. As a complementizer (that):
 rAmudu pallu wiMtAdani mohana ceVppAdu.
 ‘Ram’ ‘fruits’ ‘eat_will’ ‘-that’ ‘mohan’ ‘told’.
 Mohan told that Ram eats fruits.
- b. As a subordinating conjunct:
 rAmudu wanani vellamannAdani mohana vellipoyAdu.
 ‘Ram’ ‘him’ ‘to_go’ ‘-told’ ‘-because’ ‘mohana’ ‘went’.
 Mohan went because Ram told him to go.

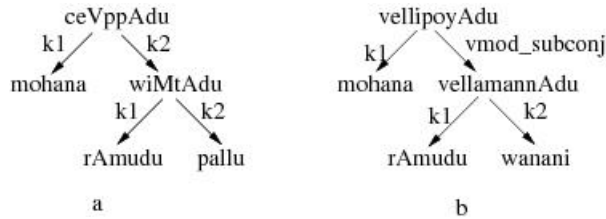


Fig. 8. Ani Constructions

The following example and its dependency structure shown in Figure 9 covers almost all the above mentioned cases:

rAmudu maMci bAludani, raGu puwakAlu caxuvuWAdani sIwa ceVpwuMxi.
 ‘Ram’ ‘good’ ‘boy’ ‘-that’ ‘(and)’ ‘Raghu’ ‘reading’ ‘books’ ‘-that’ ‘sIwa’
 ‘says’.
 Sita says that Ram is a good boy and reads Raghu’s books.

5 Parsing Issues

In this section, we shall look at how the above discussed cases will be problematic in parsing.

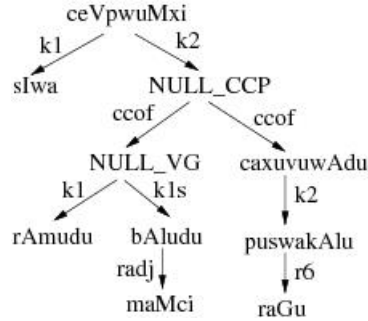


Fig. 9. All Cases

5.1 Genitives

As discussed earlier due to the potentiality of dropping the genitive marker, the sentences become ambiguous.

raGu puswakaM rAmudiki iccAdu.
 ‘Raghu’ ‘book’ ‘Ram’ ‘-to’ ‘gave’.

The above sentence is ambiguous. It’s two different interpretations are shown below.

- a. Raghu’s book was given to Ram [by somebody].
- b. Raghu gave a book to Ram.

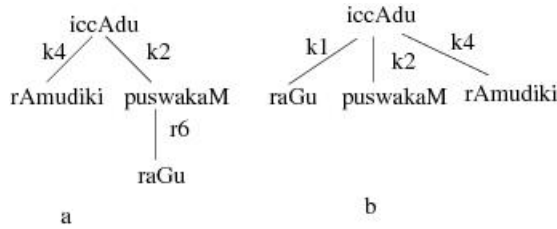


Fig. 10. Genitive

At the sentence level we cannot predict the sense among the two. It needs contextual reasoning. Hence, for a grammar-driven approach it will be wise to generate the possible two parses as shown in the Figure 10. Later a prioritizer will select the most appropriate parse based on relevant contextual features [5].

5.2 Conjuncts

For suffixes, a transformation frame [6] for the TAM, ‘we’, corresponding to the verb frame [3] for the verb *vellu* repairs the dependency tree. For lexical items, the demands

will be met by the conjunct frame¹⁶. But there is a problem in identifying the conjunction in case of Dheergas. The main problem here is "how to insert a NULL_CCP" in the sentence. A heuristic to resolve the problems:

If the vowel elongated lexical item and the word¹⁷ succeeding to it are of the same POS category insert a NULL_CCP between those words and this proves to be true for 98.6% cases.

In the example given, rAmudu and sIwa, both are proper nouns and hence NULL_CCP is inserted. But in the sentence "rAmudu (Ram) kApI (coffee) wAgAdu (drank)" it won't be inserted because rAmudu is a proper noun whereas kApI is a common noun. A coordinating conjunct preserves the category of the words it conjoins.

5.3 Copula

The main problem here is "how to insert a NULL_VG" in the sentence. Below we state two possible heuristics to overcome this problem. Insert a NULL_VG if:

1. There is no main verb in the sentence.
2. If there is a clause end marking lexical item, like *ani*, and if that clause doesn't contain any verb. This heuristic fails if we consider the free word orderness of the language.
3. Once NULL_VG is inserted, we can check for proximity to identify k1 and k1s which are the potential children for NULL_VG, though there are cases where this fails too.

We need to address this issue in detail (more importantly, verb missing in a clause) as the above heuristics does not work all the time.

6 Conclusion and Future Work

In this paper we have introduced an ongoing effort to annotate Telugu sentences with dependency relations. We stated the motivation behind following the Paninian framework in the Indian language scenario. We discussed different cases where we came up with some generalizations for annotations. We also showed and discussed why and how these cases are problematic in parsing in perspective of a grammar-driven approach. In the future our major goal is to increase the number of annotated sentences in the treebank.

Along with that we wish to start exploring the treebank in terms of understanding which features of the language play a vital role in parsing from the perspective of Machine Learning. We are trying to adopt a two-stage constraint based parsing architecture for Telugu.

¹⁶ See [4], [5] for details on the conjunct frames and how they are handled in a two stage parsing architecture.

¹⁷ Words and lexical items are used interchangeably.

Acknowledgement

We would like to thank Ganga Bhavani, D V Sriram, Phani Gadde, Bharat Ambati for developing parts of the treebank.

References

1. Bamman, D., Crane, G.: The design and use of a Latin dependency treebank. In: Proc. of TLT 2006, pp. 67–78. FAL MFF UK, Prague (2006)
2. Begum, R., Husain, S., Dhvaj, A., Sharma, D., Bai, L., Sangal, R.: Dependency annotation scheme for Indian languages. In: Proc. of IJCNLP 2008 (2008)
3. Begum, R., Husain, S., Sharma, D.M., Bai, L.: Developing Verb Frames in Hindi. In: Proc. of LREC 2008, Marrakech, Morocco (2008)
4. Bharati, A., Husain, S., Sharma, D.M., Sangal, R.: A Two-Stage Constraint Based Dependency Parser for Free Word Order Languages. In: Proc. of the COLIPS IALP 2008, Chiang Mai, Thailand (2008)
5. Bharati, A., Husain, S., Sharma, D.M., Sangal, R.: In: Proc. of IWPT 2009, Paris (2009)
6. Bharati, A., Chaitanya, V., Sangal, R.: Natural Language Processing: A Paninian Perspective, pp. 65–106. Prentice-Hall of India, New Delhi (1995)
7. Bhatt, R., Narasimhan, B., Palmer, M., Rambow, O., Sharma, D.M., Xia, F.: A Multi-Representational and Multi-Layered Treebank for Hindi/Urdu. In: Proc. of TLT 2009 (2009)
8. Brants, S., Dipper, S., Hansen, S., Lezius, W., Smith, G.: The TIGER Treebank. In: Proc. of TLT 2002 (2002)
9. Bosco, C., Lombardo, V.: Dependency and relational structure in treebank annotation. In: Proc. of Workshop on Recent Advances in Dependency Grammar at COLING 2004 (2004)
10. Hajicova, E.: Prague Dependency Treebank: From Analytic to Tectogrammatical Annotation. In: Proc. TSD 1998 (1998)
11. Hudson, R.: Word Grammar. Basil Blackwell, 108, Cowley Rd, Oxford, OX4 1JF, England (1984)
12. Krishnamurti, B., Gwynn, J.P.L.: A grammar of modern Telugu. Oxford University Press, Delhi, New York (1985)
13. Mel'cuk, I.A.: Dependency Syntax: Theory and Practice. State University Press of New York (1988)
14. Marcus, M., Santorini, B., Marcinkiewicz, M.A.: Building a large annotated corpus of English: The Penn Treebank. In: Computational Linguistics (1993)
15. Rambow, O., Creswell, C., Szekely, R., Taber, H., Walker, M.: A dependency treebank for English. In: Proc. of LREC 2002 (2002)
16. Shieber, S.M.: Evidence against the contextfreeness of natural language. *Linguistics and Philosophy*, 8, 334–343 (1985)
17. van der Beek, L., Bouma, G., Malouf, R., van Noord, G.: The Alpino dependency treebank. In: Computational Linguistics in the Netherlands (2002)
18. Vaidya, A., Husain, S., Mannem, P., Sharma, D.M.: A karaka-based dependency annotation scheme for English. In: Gelbukh, A. (ed.) *CICLing 2009*. LNCS, vol. 5449, pp. 41–52. Springer, Heidelberg (2009)