Clause Boundary Identification for Tamil Language Using Dependency Parsing

R. Dhivya[1], V. Dhanalakshmi[2], M. Anand Kumar[1], and K.P. Soman[1]

[1] Centre for Excellence in Computational Engineering and Networking,
Amrita School of Engineering, AmritaVishwaVidyapeetham, Coimbatore, India
[2] Department of Tamil, SRM University, Kattankulathur
{r.dhivya23,dhanagiri}@gmail.com,
anandkumar@yahoo.co.in, kp_soman@amrita.edu

Abstract. Clause boundary identification is a very important task in natural language processing. Identifying the clauses in the sentence becomes a tough task if the clauses are embedded inside other clauses in the sentence. In our approach, we use the dependency parser to identify the boundary for the clause. The dependency tag set, contains 11 tags, and is useful for identifying the boundary of the clause along with the identification of the subject and object information of the sentence. The MALT parser is used to get the required information about the sentence.

Keywords: Natural Language Processing (NLP), Dependency parser, Clause boundary, Parts of Speech (POS), Shift reduce parser, MALT.

1 Introduction

Clause boundary identification plays a major role in various NLP tasks. If the sentence length is too long, it is split into many simple clauses, which makes the translation process an easy one. If the sentences are connected using connectives or separated by comma, the clauses can be split easily. In other cases, the process is tedious. So, the dependency parser is used, to identify the boundaries of the clauses, using the MALT tool.

2 Related Works

Clause boundary identification is a significant work in NLP tasks in the last few years. R. Vijay Sundar Ram and SobhaLalithaDevi[1] identified the clause boundary using the Conditional Random Fieldsand used a hybrid approach. Fredrik Jorgensen[2] detected the clause boundaries by classifying coordinating conjunctions in spoken language discourse as belonging to either the syntactic level or discourse level of analysis. Hyun-Ju Lee et al. [3] proposes a method for Korean clause boundary recognition by recognizing the ending points of clauses first, and then identify the starting points by considering the typological characteristics of Korean.

V.V. Das, E. Ariwa, and S.B. Rahayu (Eds.): SPIT 2011, LNICST 62, pp. 195–197, 2012.

Tomohiro Ohno et al. [4]proposed a technique for clause-by-clause basis by identifying the clauses based on clause boundaries analysis, analyzes the dependency structures of them, and tries to decide the dependency relations with another clause. Tomohiro Ohno et al. [5]identified clause boundaries in two levels, one is the clause level and the other one is the sentence level. Dan Lowe Wheeler[6] presented a paper for machine translation through clausal syntax for his thesis, which is a tree to tree translation from English to Chinese that predicts the English clause structure from Chinese clause structure.

3 Proposed Method

The input sentences are first tokenized and given to the POS tagger and chunker. It is then preprocessed to the parser input format and then passed to the parser. The parsed output is then converted to a digraph format which serves as the input to the tree viewer. The tree viewer generates the tree structure forthe parser output. The block diagram for the parsing system is given in Figure 1.

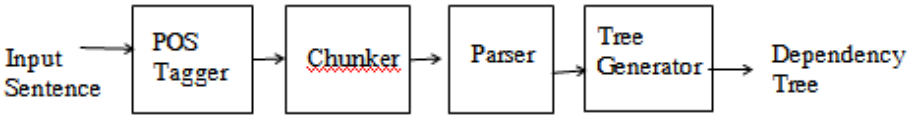

Fig. 1. Block diagram of the proposed method

4 Dependency Tag Set

The tag set developed for the Dependency Parsing has 11 tags, which is listed in Table 1.

Table 1. Dependency Tagset

S.No.	Tag	Description
1	<ROOT>	Head Word
2	<N.SUB>	Nominal Subject
3	<D.OBJ>	Direct Object
4	<I.OBJ>	Indirect Object
5	<NST.MOD>	Spatial Time Modifier
6	<CL.SUB>	Clausal Subject
7	<CL.DOBJ>	Clausal Direct Object
8	<CL.IOBJ>	Clausal Indirect Object
9	<SYM>	Symbols
10	<X.CL>	Clause Boundary
11	<X >	Others

5 Tool Used for Dependency Parsing

The tool used for Dependency Parsing is the MALT Parser Tool[7].In our model, we have considered the following features: word index, word, POS tag, chunk tag, dependency head and dependency relation.The word index, word, POS tag and the chunk tag are in the first four columns. The dependency head and the dependency relation should be in the 7^{th} and 8^{th} column respectively. The rest of the features are marked '_'. The output is given to the GraphViz tool to generate the tree structure.GraphViz is an open source tool that generates the tree structure from a digraph file.

6 Results and Conclusion

The parser is trained using the sentences of different patterns collected from various Tamil grammar books and so the training data covered almost all the patterns available for simple sentences and limitedcomplex sentences. The developed model showed better accuracy for the sentences of smaller length.So, including the complex sentences in the training data will improve the accuracy of the clause boundary identification.

References

1. Ram, R.V.S., Lalitha Devi, S.: Clause Boundary Identification Using Conditional Random Fields. In: Gelbukh, A. (ed.) CICLing 2008. LNCS, vol. 4919, pp. 140–150. Springer, Heidelberg (2008)
2. Jorgensen, F.: Clause Boundary Identification in Transcribed Spoken Language. In: Nivre, J., Kaalep, H.-J., Muischnek, K., Koit, M. (eds.) NODALIDA Conference Proceedings, pp. 235–239 (2007)
3. Lee, H.-J., Park, S.-B., Lee, S.-J., Park, S.-Y.: Clause Boundary Recognition Using Support Vector Machines. In: Yang, Q., Webb, G. (eds.) PRICAI 2006. LNCS (LNAI), vol. 4099, pp. 505–514. Springer, Heidelberg (2006)
4. Ohno, T., Matsubara, S., Kashioka, H., Kato, N., Inagaki, Y.: Incremental Dependency Parsing of Japanese Spoken Monologue Based on Clause Boundaries. In: Proceedings of 9th European Conference on Speech Communication and Technology, pp. 3449–3452 (2005)
5. Ohno, T., Matsubara, S., Kashioka, H., Maruyama, T., Tanaka, H., Inagaki, Y.: Dependency Parsing of Japanese Monologue Using Clause Boundaries. Springer (2007)
6. Wheeler, D.L.: Machine Translation through Clausal Syntax: A Statistical approach for Chineese to English, Technical Report, Massachusetts Institute of Technology (2008)
7. MALT Parser website, `http://www.maltparser.org/userguide.html`