Malayalam Clause Boundary Identifier: Annotation and Evaluation

Sobha, Lalitha Devi
AU-KBC Research Centre
MIT Campus, Anna University
Chennai-600044
sobha@au-kbc.org

Lakshmi, S
AU-KBC Research Centre
MIT Campus, Anna University
Chennai,600044

Abstract

Clause boundary identification has a significant role in improving the performance of different practical NLP systems. In this paper we have dealt with automatically identifying various types of clausal structures in Malayalam, a Dravidian language. The clausal sentences were collected from tourism and health domain available in the Web. We discuss about the annotation schema and the inter-annotators agreement for various clauses and also the automatic identification of clause boundaries using CRFs a Machine learning approach. To smooth the errors obtained from the CRFs tagging, we have used linguistic rules. For Inter-annotators agreement we have used kappa coefficient as the agreement statistic. The evaluation gave encouraging result.

1 Introduction

Clause identification is a shallow parsing task, where the boundaries of a clause are determined. Syntactic structure information, given by the clause boundaries in a sentence helps in improving the NLP applications. Incorporation of clause boundary identification enhances the performance of various applications such as machine translation, text-to-speech, information extraction, question answering system since it gives a deeper level of syntactic information. In English, clause is defined as a word sequence which contains a subject and a predicate. This subject can be explicit or implied. In automatically identifying the clause, the boundaries of the clauses in a sentence are marked. In the present work, we give in detail the annotation and the automatic identification of clause boundary in Malayalam. We use the machine learning approach CRFs for identification of clause boundary and use linguistic rules to correct the errors obtained from the CRF engine.

Many learning approaches were used in the clause identification task. Hidden Markov Models, Memory-based Learning, Boosting are some of them. Eva Ejerhed (1988) developed a basic clause identification system, for text to speech system to find basic surface clauses in unrestricted English text, using various combinations of finitary and stochastic methods. Leffa (1998) used a rule based method where clauses can be ultimately reduced to a noun, an adjective or an adverb regardless of their length or the number of embedded clauses they may contain. After the rule based techniques, machine learning approaches and hybrid approaches where used in clause boundary identification.

Orasan (2000) used a hybrid method for clause splitting in unrestricted English texts which used a machine learning algorithm and a shallow rule-based module. Molina et al. (2001) has used a specialized HMM approach to clause identification task where clause start and end tags were detected along with embedded clause detection. The clause identification was the shared task in CoNLL-2001 (Tjong et al., 2001).Conditional random fields is used for most of the sequence labeling tasks, such as shallow parsing by Sha (2003),named entity recognition task by McCallum et al., (2003).A multilingual method for clause splitting was done by Georgiana Puscasu (2004). Here she used the information of coordination and subordination with machine learning technique. Clause spitting was done by using conditional random fields' technique by Nguyen et.al. (2007). Alegria et al. (2008) identified systems for syntactic chunking and clause identification for Basque, combining rule-based grammars with filtering-ranking perception. An approach was presented where CRF and linguis-

The 4th Workshop on South and Southeast Asian NLP (WSSANLP), International Joint Conference on Natural Language Processing, pages 83–90,
Nagoya, Japan, 14-18 October 2013.

tic rules were used and cascaded by an error analyzer by Vijay et al., (2008).

There have been limited efforts on clause identification for Indian languages. One such approach is in Tamil where Vijay et al. (2009) has used CRFs based approach, where the syntactic rules are used for error correction. Daraksha Parveen et al. (2011) has done clause boundary identification task for Urdu using classifiers. They have used the machine learning technique, which has linguistic rules as features, to identify the clausal boundaries first and the misclassified clause boundaries were corrected using additional linguistic rules. Aniruddha Ghosh et al. (2010) had worked for Bengali, where they have used rules for identifying the clause boundaries and CRFs to classify the clause. In Bengali, corpus used is from NLP Tool Contest: ICON 2009, they have annotated the clause information. Hindi clause boundary information is automatically tagged using the Stage 1 parser for Hindi developed by Husain et al. (2009).

The paper is arranged as follows, in the next section we discuss about the various clausal structures in Malayalam. In the third section, we have explained about the annotation of clause boundary markers and explained the inter-annotator agreement. The automatic clause boundary identification system and its evaluation are explained in fourth section and finally the conclusion and reference section is presented.

2 Clause Structures

We describe in detail about the clauses and the clausal structures in Malayalam, one of the South Dravidian language. It is a verb final language and also a free word order language. It has postpositions, the genitive precedes the head noun in the genitive phrase and the Complementizer follows the embedded clause. It is a nominative-accusative language like the other Dravidian languages. Here, due to rich inflection, the morphology of the words contributes more information than in English which is a non-inflected language. The clause boundaries are indicated by suffixes attached to the verb. We have considered the following clauses for analysis, Relative participle clause (RP), Conditional clause (CON), Infinitive clause (INF), Complementizer (COM) and Main clause (MCL). The clause is identified by the type of non-finite verb present in the sentence. Different structures in each clause are described below.

2.1 Relative Participle Clause

The relative participle clause is identified by the relative participle suffix 'a' attached to the non-finite verb in a sentence. The relative participle (RP) verb takes all tense forms. The future form is essentially restricted to certain types of written usage (Asher and Kumari, 1997).Based on the constituents that follow the relative participle verb, the relative participle clause can have the following patterns.

RP verb followed by Noun

1. *da:hajalam kontuvarunna pla:stikk*
DrinkingWater bring+present+rp plastic
kuppikal vanapAlakar thatayunnunt.
bottle+pl forestguard+pl prevent+pres+be+pres
(Forest guards are preventing plastic bottles in which drinking water is brought.)

This is the most common structures of RP clause, RP verb followed by a noun phrase, which will take all the case markers. This NP can also be preceded by a genitive noun. In the above sentence RP verb ' kontuvarunna ' (bringing) is followed by a Noun phrase ' pla:stikk kuppikal'(plastic bottles)

RP verb followed by PSP

2. *tibattil ethunna ya:thrakka:r*
Tibet+loc reach+present+RP traveller+pl
6 divasam kont kaila:sa ma:nasasarovara
6 days psp kailasa manasa sarovar
darSanam natathiya shesham athirthiyil
worship did+RP psp border+loc
thirichethi curam katakkunnu.
come-back hairpin-bend cross+pres
(Travellers who reach Tibet ,finish the worship of kailasa manasa sarovar in 6 days and then come back to the border and cross the hairpin-bends.)

RP verb can also be followed by a PSP without NP in between. PSP's such as 'shesham','muthal' etc will follow the RP verb. In above sentence RP verb 'natathiya' is followed by PSP 'shesham'.

RP verb followed by a Noun and Adv

3. *nura patayunna thirama:lakalkk pinna:le*
Foam foam+present+RP wave+pl+dat behind
kuññunnal otum.
children run+future.
(Children run behind the waves which is foaming.)

In this structure of RP clause, RP verb is followed by a dative noun and an adverb. In the above given sentence ' patayunna ' is the RP verb followed by dative noun thiramalakalkk and adverb ' pinna:le '.

RP verb followed by a pronoun

4. *ra:vile kulakkatavil kulikka:n ettiyavar*
morning pond bath+inf reach+RP+pron
oru bhīkarajīvi nataththunna puja kant
one monster perform+RP worship see
bhayann a:lukale vilichukutti
frighten+past people call.

(In the morning those people who came to have a bath in the pond got frightened seeing one monster performing worship and they went and called others.)

The RP verb can be followed by a pronoun, similar to RP verb followed by NP. Here the pronoun can be agglutinated with the RP verb. In the above sentence ' ettiyavar ' (those who reached) is the RP verb followed by a 3 person plural epicene pronoun.

2.2 Infinitive Clause

Infinitive (INF) verb does not take tense markers and the infinitive marker is 'a:n'. The infinitive clause in the sentence is identified using the infinitive verb.

5. *a:nakkuttannale ka:na:n e:rravum kututhal*
Herds of Elephants see+INF utmost
avasaram kittunna pradesam
opportunity get+RP place
añchuna:zhikathot vanama:nu
Anchuna:zhikathot is forest
(Anchuna:zhikathot forest is the place where you get utmost opportunity to see herds of elephants.)
Here the infinitive verb is ' ka:na:n ' ' to see ' .
INF+Inclusive marker

6. *ka:ppa:t gramaththilute ozhukunna*
Kappat village+through(case) flow+past+RP
korappuzhayilute bott sava:ri nataththa:num
korappuzha boat riding perform+RP+inc
ka:ylil po:kanum saukaryamunt
backwater go+RP+inc has facility
(There is facility to do boat riding and also go to the backwaters in the korappuzha river flowing through kappat village.)

Here ' nataththa:num ' and ' po:kanum ' is the 2 Infinitive verbs.

2.3 Conditional Clause

Conditional clause is identified by conditional (CON) verb. The suffixes for conditional verb are 'a:l'.
CON verbs take tense markers. It occurs in present, past and future tense.

7. *ivite ninnu 3 manikkur trekkin nataththiya:l*
here from 3 hours trekking do+cond
varya:ttumottayil varaya:tukale
Varayattumotta+loc Nilgiri tahr
ka:na:m
see+future+mod

(From here if we do trekking for 3 hours (we) can see Nilgiri tahr from Varayattumotta.)

The conditional verb is “nataththiya:l” with the suffix “a:l”. Here the embedded clause is “ivite ninnu 3 manikkur trekkin nataththiya:l” “If we do trekking for 3 hours”. Main clause is “varya:ttumottayil varaya:tukale ka:na:m ” “can see Nilgiri tahr from Varayattumotta”.
The addition of an emphatic particle 'ee' to a conditional form also occurs to express the concept of 'only if'.

8. *nurrant tha:ntiya parvathavantiyil*
Century cross+past+RP mountain vehicle+loc
mala kayariya:le uutti ya:thra
hill climb+con+emphatic ooty trip
purnnamaku
fullfilled
(Your ooty trip will be fullfilled only if you climb the hill in the centuries old mountain train.)

Here ' kayariya:le ' is the conditional verb with the emphatic particle.
Unfulfilled conditions are marked using the suffix 'enkil'. (Asher and Kumari, 1997)

9. *i: varsśaththe rathhothsavam ka:nanamenkil*
this year+acc chariot festival see-want+cond
navambar 8 muthal 16 vareyulla divasannalil
november 8 from 16 upto day+pl+loc
kalppa:ththi sandarsichcha:l mathi.
Kalppathi visit enough
(If u want to see the radhostav this year u can visit kalpathi from November 8 to 16.)

Here ' ka:nanamenkil' is the unfulfilled conditional verb.

2.4 Complementizer Clause

'ennu' is the Complementizer (COM) marker in Malayalam, which is similar to 'that' in English. The Complementizer Clause can occur in three

different positions in a sentence. It can be before, after or within the main clause.

10. *ra:man varum enn kutti paraññu*
Raman come+fut that child tell+past
(The child told that Raman might come.)

11. *kutti paraññu ra:man varum enn.*
Child tell+past raman come+fut that.
(The child told that Raman might come.)

12. *kutti raaman varum ennu parannu.*
child Raman come+fut that tell+past
(The child told that Raman might come.)

Out of the three the third form is most common in the tourism and health corpus which we had selected for annotation.

13. *ka:na:mpuzha ozhukiyirunna kanththur gramam*
kanampuzha flowing+RP ka:nathur village
pinnīt kannur enna peril ariyappettu enn oru
later Kannur name known that one
abhiprayam
opinion
(There is an opinion that Kanathur village through which kanampuzha river was flowing later came to be known as Kannur.)

"pinnīt kannur enna peril ariyappettu" is the complementizer clause in the above sentence.

3 Annotation and Inter-annotator Agreement

3.1 Corpus

The clause tagged corpus used in the CoNLL Shared task 2001, is one of the first available corpus. In that corpus, they have used "S*" to indicate clause start and "*S" for indicating clause end. The corpus was presented in column format, which has word, part-of-speech, chunk and the clause boundary tags. In this style of annotation they had only marked the start and end of the clauses and the type of clause is not mentioned. In our tagging schema we have tagged the type of clauses as well as the start and end of the clause. We selected about 6415 tourism and 385 health corpus sentences from the Web and training set consisted of 5000 sentences from both the domains. Testing of the system was done with 401 unseen sentences from the tourism corpus.

3.2 Annotation

The sentence boundaries were not given in the preprocessed data. We have identified relative participle clause, conditional clause, infinitive clause, complementizer and main clause. We have used the tags {RP} and {/RP} for RP clause start and end respectively. Similarly we have used the following tags to represent the start and end tags, {INF} and {/INF} for INF clause, {CON} and {/CON} for CON clause, {COM} and {/COM} for COM clause and {MCL} and {/MCL} for main clause.

14. *{CON} ivite ninn 3 manikkur*
here from 3 hours
trekkin natathithiya:l {/CON}
trekking do+cond
{MCL} varya:ttumottayil varaya:tukale
Varayattumotta+loc Nilgiri tahr
ka:na:m {/MCL}
see+future+mod
(From here if we do trekking for 3 hours (we) can see Nilgiri tahr from Varayattumotta.)

Above example shows how the annotation is done using our schema.

3.3 Inter- annotator Agreement

We have measured the inter-annotators agreement as there could be ambiguity in identifying the start and end of clauses. Inter-annotator agreement is the degree of agreement among annotators. It is the percentage of judgments on which the two analysts agree when coding the same data independently. There are different statistics for different types of measurement. Some are joint-probability of agreement, Cohen's kappa and the related Fleiss' kappa, inter-rater correlation, concordance correlation coefficient, Cochran's Q test, intra-class correlation and Krippendorff's Alpha. We use Cohen's kappa as the agreement statistics. The kappa coefficient is generally regarded as the statistic of choice for measuring agreement on ratings made on a nominal scale. It is relatively easy to calculate, can be applied across a wide range of study designs, and has an extensive history of use. The kappa statistic k is a better measure of inter-annotator agreement which takes into account the effect of chance agreement (Ng et al., 1999).

$k = (p_o - p_c)/(1- p_c)$

where p_o is agreement rate between two human annotators and p_c is chance agreement between two annotators. The results of kappa-like agreement measurements are interpreted in six categories as follows (Yalçınkaya et al., 2010). Kappa Score Agreement

Score	Agreement
<0	Less than chance agreement
0.0–0.2	Slight agreement
0.2– 0.4	Fair agreement

0.4–0.6	Moderate agreement
0.6–0.8	Substantial agreement
>0.8	Almost perfect agreement

We calculated the kappa score for each clause start and end and are presented in the following table.

Clauses	Start	End
RP	0.89	0.82
CON	1	1
COMP	0.63	0.63
INF	1	1
MCL	0.84	0.89

Table 1. Kappa Score

As the clause end is actually identified during the clause boundary identification task the level of agreement between the annotators should be more for clause end. But when we see the RP clause the clause end agreement is low compared to clause start. The complementizer clause is having only substantial agreement. The overall kappa score is 0.87 that means it is almost perfect agreement between the annotators.

4 Automatic Clause Boundary Identifier

We have used a hybrid method for identification of the clause boundaries, a machine learning method CRFs for the detection of the boundaries of the clause and the type of clause and linguistic rule based approach to correct the error made by the CRFs approach. The input sentences are preprocessed for sentence splitting, tokenizing, morphological analysis, part-of-speech tagging (POS) and chunking. The features are obtained from the linguistics analysis of the various types of clauses which are discussed in detail in section 4.2.The error analysis uses syntactic rules specific to certain constructions where there are difficulty in identifying the start or end by the ML method. In the following sections we give in detail both the approaches.

4.1 Conditional Random Fields (CRF)

CRFs are an undirected graphical model. Here conditional probabilities of the output are maximized for given input sequence (Lafferty, 2001). This technique is used for various tasks in NLP.

Learning: Given a sample set X containing features $\{X_1,.....,Xn\}$ along with the set of values for hidden labels Y i.e. clause boundaries $\{Y_1,.....,Y_n\}$, learn the best possible potential functions.

Inference: For a given word there is some new observable x, find the most likely clause boundary y* for x, i.e. compute (exactly or approximately):

$y^*=\text{argmax}_y P(y|x)$

Linear-chain CRFs thus define the conditional probability of a state sequence given as

$$P_\Lambda(\mathrm{s}|\mathrm{o}) = \frac{1}{Z_\mathrm{o}} \exp\left(\sum_{t=1}^{T}\sum_{k=1}^{K} \lambda_k f_k(s_{t-1}, s_t, \mathrm{o}, t)\right),$$

where Z_o a normalization factor over all state sequences, fk (s_{t-1},s_t,o,t) − is an arbitrary feature function over its arguments, and λ_k (ranging from $-\infty$ to ∞) is a learned weight for each feature function. A feature function may, for example, be defined to have value 0 in most cases, and have value 1 if and only if s_{t-1} is state #1 (which may have label OTHER), and s_t is state #2 (which may have START or END label),and the observation at position t in o is a relative pronoun or a conditional marker. Higher λ weights make their corresponding FSM transitions more likely, so the weight λ_k in the above example should be positive since the word appearing is any clause marker (such as conditional or relative clause marker) and it is likely to be the starting of a clause boundary. Here we have used CRF++ tool which is available on the web (Kudo, 2005).

4.2 Features

The performance of the machine leaning technique depends on the features used in learning. The features used in our approach are word, pos, chunk, morph information and the clause type information. The type of clause is determined by the suffix the non-finite verb takes and this is used as one of the features. The chunk boundaries are more important as a feature since most of the start and end boundaries of the clause matches with that of the chunk boundaries. Part-of-speech information provides the context and definition of the words in a sentence.

Consider example 1

da:hajalam kontuvarunna pla:stikk
DrinkingWater bring+present+rp plastic
kuppikal vanapAlakar thatayunnunt.
bottle+pl forestguard+pl prevent+pres+be+pres
(Forest guards are preventing plastic bottles in which drinking water is brought.)

In the above clause tagged sentence the word "kontuvarunna" is the RP verb and its POS is VM and chunk information is B-VGNF and morph information is v+rp. These information is used by the ML approach to learn the patterns.

4.3 Error Analyzer

The error analyzer module is used for detecting the erroneous clause boundary markings done by CRFs module. For example consider the below given sentence.

15. *athinu sesśam athuvare uuzham kaththu*
after that till then opportunity wait
nilkkunna komarannalum janannalum onnichch
stand+RP komaram+pl and people+pl together
a:vesalahariyote dikkukal muzhannunna
excitement in all directions echoing+RP
tharaththil marankampu kont kśśethraththinre oot
kind wooden stick with temple bell-metal
meññna melkkurayil atichchu kont munnu
thatched+RP upper roof beating psp three
pravasyam valamvekkunnu
times rounds

(After that those komarams and people waiting for an opportunity together in excitement, rounds the temple 3 times while beating the bell-metal thatched upper roof with a wooden stick in an echoing sound .)

From the output generated for the above sentence it was noticed that RP clause ending was not marked at *"athinu sesśam athuvare uuzham kaththu nilkkunna komarannalum"* and RP start was not marked at "*janannalum onnichch a:vesalahariyote dikkukal muzhannunna tharaththil"*. For detecting the errors, the training data (gold standard data) itself is given as test data to CRFs system. The erroneous clause boundary marked sentences which are filtered out by the error analyzer module are further analysed using linguistic rules. The error patterns derived by processing the gold standard data are compared with the output of the CRFs module to detect the incorrect clause boundaries marked by the CRFs module.

4.4 Grammatical Rules

The rules are used to treat the erroneous clause boundary markings done in the CRFs module. The rules are used to identify the unidentified clause boundaries in the given sentences. It actually fine tunes the CRFs output. The grammatical rules used in the clause boundary identification work are as follows.

To get the relative participle clause boundary

If the current token is a PSP, the previous is a RP verb then current PSP is the probable RP clause end.

-1 VM+RP
0 PSP=1 RP clause end

If the current token is NP and the previous one was a relative participle verb and if the next token is not a PSP then the current token becomes the RP clause end.

-1 VM+RP
0 NP RP clause end

To get the conditional clause boundary

If the current verb has a conditional marking suffix, then the current verb is marked for probable conditional clause end.

0 VM+CON = 1 CON clause end

To get the infinitive clause boundary

If the current verb has the infinitive suffix then the current verb is marked for probable infinitive clause end.

0 VM+INF=1 INF clause end

To get the complementizer clause boundary end

If the current word is a complementizer "ennu ", the previous word is a finite verb followed by a noun phrase. The COM clause end boundary is marked.

-1 VGF
0 Complementizer=1 COM clause end
1 NP

Once these rules are run, the probable clause start positions are marked based on the probable clause ends marked.

For the example 14 when Boolean entries which obey the linguistic rules was given as a column in training data the output was free of errors.

4.5 Evaluation and Discussion

There were 358 relative participle clauses, 36 conditional clauses, 7 complementizer clauses and 264 main clauses in the testing sentences. We measured the performance in terms of precision and recall and F-measure, where precision is the number of correctly recognized clauses to the number of clauses marked in the output, recall is the number of correctly recognized clauses to the number of clauses and F-measure is the weighted harmonic mean of precision and recall.

Clauses	Recall	Precision	F-measure
RP open	96.65	89.87	93.14
RP close	69.51	73.1	71.3
CON open	54.3	100	70.4
CON close	80.6	100	89.26
MCL open	58.46	52.77	55.47
MCL close	90.9	89.22	90.05
INF open	33.33	40	36.36
INF close	66.67	80	72.73
COM open	57	100	72.6
COM close	100	100	100

Table 2. Performance of the system

On analyzing the performance tables, it is clear that the propagation of errors from the prior modules affect the performance, as this identification tasks requires all the three analysis, morph analysis, POS and chunk information to be correct, to introduce the tag at the correct chunk.
Consider example below

16. thotine kavachchuvaykkunna
Stream+acc straddle+pres+RP
cheriya palam katann valaththott thiriññ
small bridge go right turn
kayariya:l irumpunnikkara vare nattu vazhi
Ascend+cond irumpoonnikkara till land

(If you turn right and ascend through the small bridge straddling through the stream there is land till irumpunnikkara.)
Here there is an RP clause followed by a conditional and main clause. But the output of the system tags RP clause properly, but the Conditional clause start is not marked properly.

17. *annu rajkumari paraññ*a *kathhakal*
That day princess tell+past+RP stories
kett pamp pirrenn thirichchupoyennan aithihyam
listen snake next day went back history
(The history is that the snake went back after hearing the story told by the princess.)

In such type of sentence formation the RP clause end was not properly marked.

5 Conclusion

In this paper, we have discussed about some clausal structures in Malayalam, described about annotation of clause boundaries in Malayalam sentences. Finally we have explained about the automatic clause boundary identifier using CRFs. We have discussed about the factors affecting the identification task. The system can be further improved with more rules.

References

, A. and Uria, L. 2008. Chunk and clause identification for Basque by filtering and ranking with perceptrons. *In Proceedings of SEPLN*.

Aniruddha Ghosh, Amitava Das and Sivaji Bandyopadhyay. 2010. Clause Identification and Classification in Bengali. In *Proceedings of the 1st Workshop on South and Southeast Asian Natural Language Processing (WSSANLP, 23rd International Conference on Computational Linguistics (COLING)*, Beijing, 17-25.

A. McCallum and Wei Li. 2003. Early Results for Named Entity Recognition with Conditional Random Fields, Feature Induction and Web Enhanced Lexicons. In *Proceedings of CoNLL-2003*, Edmonton Canada, pp. 188–191.

Antonio. Molina and Ferran. Pla. 2001. Clause Detection using HMM. In *Proceedings of CoNLL-2001*, Toulouse, France.

Constantin Orasan. 2000. A hybrid method for clause splitting. In *Proceedings of ACIDCA 2000 Corpora Processing*, Monastir, Tunisia, pp. 129 – 134.

D. Parveen, A. Ansari, and R.Sanyal. 2011.Clause Boundary Identification using Clause Markers and Classifier in Urdu Language,*12th International Conference on Intelligent Text Processing and Computational Linguistics CICLing*.

Eva Ejerhed. 1988. Finding Clauses in Unrestricted Text by Finitary and Stochastic Methods.In *Proceedings of the 2nd Conference on Applied Natural Language Processing*, Austin Texas, pp. 219-227.

Erik F. Tjong Kim Sang and Herv`e D`ejean. 2001. Introduction to the CoNLL-2001 shared task: clause identification. In *Proceedings of the 2001 workshop on Computational Natural Language Learning.* Toulouse, France.

Fei Sha and Fernando Pereira. 2003. Shallow parsing with conditional random fields. In *Proceedings of HLT-NAACL03*. pp. 213–220.

Georgiana Puscasu. 2004. A Multilingual Method for Clause Splitting. In *proceedings of the 7th Annual Colloquium for the UK Special Interest Group for Computational Linguistics*. Birmingham, UK. pp. 199-206.

H.T. Ng, C.Y., Lim, S.K., Foo. 1999. A Case Study on Inter-Annotator Agreement for Word Sense Disambiguation. In *Proceedings of the {ACL} {SIGLEX} Workshop on Standardizing Lexical Resources {(SIGLEX99)}*. Maryland. pp. 9-13.

Husain,S.,Gadde,P.,Ambati,B., Sharma, D.M.,Sangal, R. 2009. A modular cascaded approach to complete parsing. In *proceedings of COLIPS International Conference on Asian Language*.

R.E. Asher, and T.C. Kumari. 1996. *MalayalamRoutledge*, London and New York.

Sobha.L, and B.N., Patnaik. 2002. VASISTH-An Anaphora Resolution System for Malayalam and Hindi. *Symposium on Translation Support Systems,* IIT Kanpur.

Taku Kudo. 2005. CRF++, an open source toolkit for CRF. http://crfpp.sourceforge.net .

Vilson J. Leffa.1998. Clause processing in complex sentences. In *Proceedings of the First International Conference on Language Resource and Evaluation.* Vol 1, pp. 937 – 943.

Vijay Sundar Ram, R and Sobha, Lalitha Devi. 2008 Clause Boundary Identification Using Conditional Random Fields. In *Lecture Notes in Computer Science, Proceedings of the 9th international conference on Computational linguistics and intelligent text processing Springer*. Verlag. pp. 140-150.

Vijay Sundar Ram,R T. Bakiyavathi and Sobha. L.2009. Tamil Clause Identifier. *PIMT Journal of Research.* Patiala. Vol.2. No.1, pp. 42-46.

Vinh Van Nguyen Minh Le Nguyen and Akira Shimazu. 2007. Using Conditional Random Fields for Clause Splitting. In *Proceedings of the Pacific Association for Computational Linguistics*. University of Melbourne Australia.