World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:3, 2008

Structural Parsing of Natural Language Text in Tamil Using Phrase Structure Hybrid Language Model

Selvam M, Natarajan. A M, and Thangarajan R

***Abstract*—**Parsing is important in Linguistics and Natural Language Processing to understand the syntax and semantics of a natural language grammar. Parsing natural language text is challenging because of the problems like ambiguity and inefficiency. Also the interpretation of natural language text depends on context based techniques. A probabilistic component is essential to resolve ambiguity in both syntax and semantics thereby increasing accuracy and efficiency of the parser. Tamil language has some inherent features which are more challenging. In order to obtain the solutions, lexicalized and statistical approach is to be applied in the parsing with the aid of a language model. Statistical models mainly focus on semantics of the language which are suitable for large vocabulary tasks where as structural methods focus on syntax which models small vocabulary tasks. A statistical language model based on Tri-gram for Tamil language with medium vocabulary of 5000 words has been built. Though statistical parsing gives better performance through tri-gram probabilities and large vocabulary size, it has some disadvantages like focus on semantics rather than syntax, lack of support in free ordering of words and long term relationship. To overcome the disadvantages a structural component is to be incorporated in statistical language models which leads to the implementation of hybrid language models. This paper has attempted to build phrase structured hybrid language model which resolves above mentioned disadvantages. In the development of hybrid language model, new part of speech tag set for Tamil language has been developed with more than 500 tags which have the wider coverage. A phrase structured Treebank has been developed with 326 Tamil sentences which covers more than 5000 words. A hybrid language model has been trained with the phrase structured Treebank using immediate head parsing technique. Lexicalized and statistical parser which employs this hybrid language model and immediate head parsing technique gives better results than pure grammar and trigram based model.

***Keywords*—** Hybrid Language Model, Immediate Head Parsing, Lexicalized and Statistical Parsing, Natural Language Processing, Parts of Speech, Probabilistic Context Free Grammar, Tamil Language, Tree Bank.

Manuscript received December 27, 2007. This work was supported in part by Tamil Virtual University, Chennai, India.

Selvam M is Assistant Professor, Department of Information Technology, Kongu Engineering College, Perundurai - 638052, Erode, Tamilnadu, India. Phone: +91-4294-226570, Mobile: +91-9486655106; fax: +91-4294-220087; e-mail: amm_selvam@yahoo.co.in.

Natarajan A M is Principal of Kongu Engineering College, Perundurai – 638052, Erode, Tamilnadu, India.

Thangarajan R is Assistant Professor, in the Department of Information Technology, Kongu Engineering College, Perundurai – 638052, Erode, Tamilnadu, India. (e-mail: thangs_68@yahoo.com)

I. Introduction

PARSING is an important process of Natural Language Processing (NLP) and Computational Linguistics. It is used to understand the syntax and semantics of a natural language sentences confined to the grammar. Parser is a computational system which processes input sentences according to the productions of the grammar, and builds one or more constituent structures called parse trees which conform to the grammar. A parser permits a grammar to be evaluated against a potentially large collection of test sentences, helping the linguist to identify shortcomings in their analysis.

A. Structural Approach

In a language, group of consecutive words act as a constituent. Context Free Grammar (CFG) which is also called phrase structure grammar have been used to model constituents successfully in English. However, there are many disadvantages in using CFG for natural languages like ambiguity, left-recursion, repeated parsing of sub-trees. If a sentence is structurally ambiguous, then the grammar assigns more than one parse tree. It will be difficult to use CFG in languages that do not follow strict word order style like in English.

B. Statistical Approach

Statistical methods are primarily data driven. The frequencies of patterns as they occur in any training corpora are recorded as probability distributions. These methods mainly focus on short term relationship among words in sentences due to the N-gram hits which depend on large training set [1] and are suitable to model large vocabulary tasks. Whereas structural methods focus on syntax with long term relationship among words manifested in parse trees. Structural parsing is widely used in small vocabulary tasks. To add the structural component in statistical approach and balance the vocabulary size, Lexicalized and Statistical Parsing (LSP) can be employed.

C. Lexicalized and Statistical Parsing and its Processes

In order to overcome the problem of ambiguity, the CFG is augmented by probabilistic component. A probabilistic context free grammar (PCFG) is a CFG in which each rule is

annotated with probability of choosing that rule. PCFG probabilities can be learnt from parsing a training corpus [2][3]. Even though PCFG can resolve ambiguity by its probabilistic component, still PCFG is insensitive to words. Thus incorporating lexical information in PCFG has become important. The performance of PCFG can be further enhanced by conditioning a rule on the lexical head of its non-terminals. This is known as Lexicalized Statistical Parsing [4].

LSP has been enormously successful, but the complexity is increased. LSP is sensitive to individual lexical items and incorporation of these lexical items into features or parameters gives rise to complexity. LSP comprises pre-processing, morphological analysis, tagging, Treebank generation, building of language model and training the parser or language model. Language models are highly useful in applications like speech recognition and machine translation [5][6]. A general framework of LSP with language model is shown in figure 1.

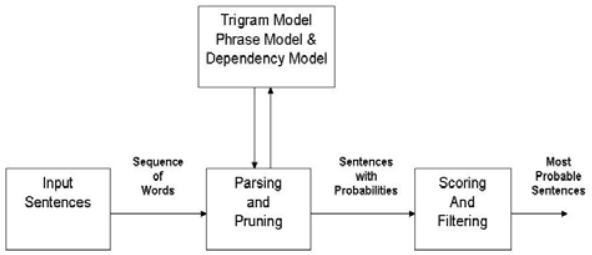

Fig 1 Framework of Lexicalized and Statistical Parser

Structural component is applied by means of Part Of Speech (POS) tagging and phrasing, construction of Treebank, and training. Language model is created with the aid of Treebank and statistical parsing is done for test sentences using the language model [7].

a. *Lexicalization*

Punctuations and special characters in sentences are removed and sentence beginning and ending markers are placed during pre-processing. POS tags are formed with morphological analysis in mind. Every word is assigned with a POS tag. Hence the POS tag and word pair forms the leaves of the parse tree of a sentence. Treebank is generated by grouping words into the phrases and constituents, and phrases into parse trees for each and every sentence of the corpus.

b. *Building of Language Model*

Language model is trained using phrase structure Treebank with immediate head parsing technique which generates trigram probabilities among head words of the constituent structures of sentences which balances syntax and semantics. This language model is hybrid in nature which contains trigram probabilities among the head words which balances memory and processing time.

c. *Statistical Parsing*

Statistical parsing is applied with the head words in the constituent structures of NL sentences and better performance is achieved [8]. This Lexicalized and Statistical Parsing with immediate head parsing technique and hybrid language model covers the advantages of free ordering of words, focus on syntax with semantics and long term relationship.

D. Features of Tamil Language

Grammar of Tamil language is agglutinative in nature. Suffixes are used to mark noun class, number and case. Tamil words consist of a lexical root to which one or more affixes are attached. Most of the Tamil affixes are suffixes which can be derivational or inflectional. The length and extent of agglutination is longer in Tamil resulting in long words with large number of suffixes.

In Tamil, nouns are classified into rational and irrational forms. Humans come under the rational form whereas all other nouns are classified as irrational. Rational nouns and pronouns belong to one of the three classes: masculine singular, feminine singular and rational plural. Irrational nouns belong to one of two classes: irrational singular and irrational plural. Suffixes are used to perform the functions of cases or post positions. Tamil verbs are also inflected through the use of suffixes. The suffix of the verb will indicate person, number, mood, tense and voice.

Tamil is consistently head-final language. The verb comes at the end of the clause with a typical word order of Subject Object Verb (SOV). However, Tamil language allows word order to be changed making it a relatively word order free language. Other Tamil language features are using plural for honorific noun, frequent echo words, and null subject feature i.e. not all sentences have subject verb and object.

To cater these challenging needs, LSP employs hybrid language model developed from phrase structured Treebank. Phrase structured Treebank is developed with Part of Speech (POS) tag set of Tamil language which needs greater coverage for all nouns, verbs, other POS and their inflections. Since Treebank construction is labor intensive, at least, a medium sized vocabulary Treebank is to be employed to train the language model.

II. LANGUAGE MODEL

Language model is the heart of the parser which provides the ways and means to predict the words and sentences confined to the patterns and grammar of a language. This is classified as statistical model which deals about semantics and structural model which deals about syntax. N-gram and Trigram models are the examples of statistical model and simple phrase structure model is the example of structural model.

A. Statistical Model

In N-gram language model, each word depends probabilistically on the n-1 preceding words. This is expressed as shown in equation (1).

$$p(w_{o,n}) = \prod_{i=0}^{n-1} p(w_i \mid w_{i-n+1}, \ldots, w_{i-1}) \qquad (1)$$

When *N* is big memory and processing power requirement is high. Good results are obtained by *N=3*. This is called tri-gram language model, where each word depends probabilistically on previous two words and is shown in equation (2)

$$p(w_{o,n}) = \prod_{i=0}^{n-1} p(w_i \mid w_{i-1}, w_{i-2}) \quad (2)$$

Trigram language model is most suitable due to the capacity, coverage and computational power [9]. For shaping the trigram model into a greater level of suitability some advanced and optimizing techniques like smoothing, caching, skipping, clustering, sentence mixing, structuring and text normalization can be applied. Through these techniques marginal improvements in perplexity can be obtained. Even though statistical model is giving better performance, proper meaning can not be derived for the compound sentences due to the tri-gram hits which capture local dependencies.

B. Structural Model

Grammar based structural model is purely rule driven approach which is suitable for small vocabulary task. The grammar is applied in the form of productions and associated probabilities. Simple phrase structure model will generate parse trees which enforce all the advantages of statistical parsing. Probabilities will disambiguate a correct parse from others. Simple structural model can overcome all the disadvantages of statistical model to some extent [10] [11].

C. Hybrid Model

Significant improvements can be achieved if structural information is applied in the statistical model [12]. Some of the examples are phrase structure and dependency structure hybrid models.

III. IMMEDIATE HEAD PARSING

LSP with immediate head parsing technique is basically lexicalized in nature which conditions probabilities on the lexical content of the sentences being parsed. All of the properties of the immediate descendants of a constituent *c* are assigned probabilities that are conditioned on the lexical head of *c* [13] [14].

For example, in Figure.2 the probability that the *S* expands into *NP PP VP* is conditioned on the head of the *VP* (**எடுக்காது** [‘eh T uh k k aa T h uh’])[1] selected from sub-heads **பஞ்சு** [‘p a eh n eh ch uh’][1] (the head of the *NP*), **தண்ணீரை** [‘T h a N N iy r ay’][1] (the head of the *PP*) and **எடுக்காது** [‘eh T uh k k aa T h uh’][1] (the head of *VP*).

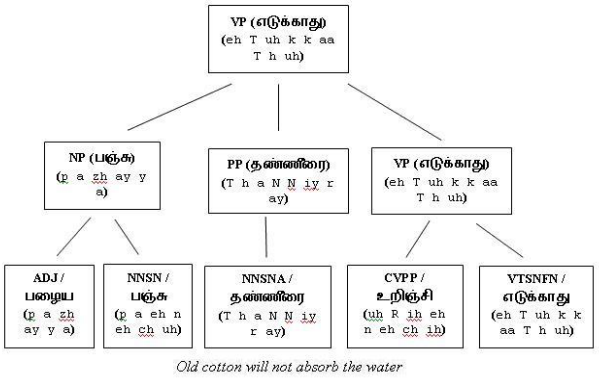

Figure 2. Parse Tree with Lexical Heads of Constituents

A. Calculating Parse Probabilities

This parsing model assigns a probability to a parse by a top-down process of considering each constituent *c* and predicting the pre-terminal *t(c)*, lexical head *h(c)* and expansion *e(c)* for each *c*. The probability of a parse is given by the equation (3)

$$p(\pi) = \prod_{c \in \pi} p(t(c) \mid l(c), H(c)).p(h(c) \mid t(c), l(c), H(c)).\ p(e(c) \mid l(c), t(c), h(c), H(c)) \quad (3)$$

where *l(c)* is the label of c (whether it is a noun phrase (*NP*), verb phrase(*VP*), etc.) and *H(c)* is the relevant history of *c*. *H(c)* approximately consists of the label, head and head-part-of-speech for the parent of *c*: *m(c)*, *i(c)*, and *u(c)* respectively. One exception is the *e(c)* distribution, where *H* only includes *m* and *u*. For simplicity equation (3) is written as shown in equation (4)

$$p(\pi) = \prod_{c \in \pi} p(t \mid l,m,u,i).p(h \mid t,l,m,u,i).p(e \mid l,t,h,m,u) \quad (4)$$

Simple hacking is done for obtaining true probabilities through right branching with a bonus multiplicative factor for constituents that end at the right boundary of the sentence, and a penalty for those that do not [15].

B. Finding Best Parse among N Parses

LSP is generative in which parser tries to find the parse of a sentence *s* defined by

$$\arg\max_{\pi} p(\pi \mid s) = \arg\max_{\pi} p(\pi, s) \quad (5)$$

Language model *p(s)* is defined by assigning a probability to all possible sentences in the language by computing the sum

$$p(s) = \sum_{\pi} p(\pi, s) \quad (6)$$

Speech-recognition systems require language models to determine words by the equation

$$\arg\max_{s} p(s \mid A) = \arg\max_{s} p(s)\,p(A \mid s) \quad (7)$$

where *A* denotes the acoustic signal, *p(s)* is the language model. The language model in equation (7) provides the priori probability to compute the posteriori probability using the class conditional probability.

IV. Development of Hybrid Language Model

Hybrid language Model is the combination of structural and statistical model. For adding the structural component POS tag is to be applied for each and every lexicon in the bottom level.

A. POS Tag-set

Parts of Speech in Tamil language take different forms and inflections as shown in table 1.

TABLE I
POS Forms and Morphological Inflections

POS & Others	Forms	Morphological inflections
Noun	Simple Noun Proper Noun Participle Noun Adjective Noun Positive Tensed Verbal Noun Negative Tensed Verbal Noun Un-tensed Verbal Noun	Number ▪ Singular ▪ Plural Gender ▪ Male ▪ Female ▪ Neutral ▪ Common ▪ Oblique
Verb	Simple Verb Transitive Verb Intransitive Verb Causative Verb Infinitive Verb Imperative Verb Reportive Verb	Person ▪ First ▪ Second ▪ Third Number ▪ Singular ▪ Plural Gender ▪ Male ▪ Female ▪ Neutral ▪ Common Tense ▪ Present ▪ Past ▪ Future Passive Honorific Negative Interrogative Suffix
Adverb	Simple Adverb	
Adjective	Simple Adjective Participle Adjective	Tense ▪ Present ▪ Past ▪ Future Negative
Preposition	Simple preposition Noun+ cases	Cases ▪ Accusative ▪ Dative ▪ Instrumental ▪ Sociative ▪ Locative ▪ Ablative ▪ Benefacive ▪ Genetive ▪ Vocative ▪ Clitics ▪ Selective Negative
Conjunction	Simple Conjunction Coordinating conjunction	Wh words ▪ What ▪ Who ▪ Whose ▪ When ▪ Where ▪ Whom ▪ Which ▪ How
	Participle	Verbal Participle Conditional participle ▪ Positive ▪ Negative
Interjection	Simple Interjection	
Others	Echo words Determiner Quantifier Complementizer Ordinal Optative	Same Different

Morphological inflections on nouns with various cases in the forms of accusative, dative, instrumental, sociative, locative, ablative, benefactive, genitive, vocative , clitics and selective can be derived [16][17]. Some other forms of verbs are transitive, intransitive, causative, infinitive, imperative and reportive. Adjective tags are generated along with tense and positive or negative participles. Prepositions are applied as the morphological inflections of nouns. Other parts of speech take simpler forms.

B. POS Tagging and Phrasing

Every sentence in the corpus is segmented into sequence of tokens and each and every token is applied with the appropriate tag for the application of direct meaning to the words. The tags and lexicons are bracketed for all the pairs. Phrasing is done among the words to form syntactic phrases and constituents. Phrases are classified based on the parts of speech as shown in table 3.

C. Phrase Structure Treebank

Phrase structure Treebank is a corpus with linguistic annotation beyond the word level. The annotation is typically a syntax tree which is manually checked and corrected [18] as shown in figure. 3.

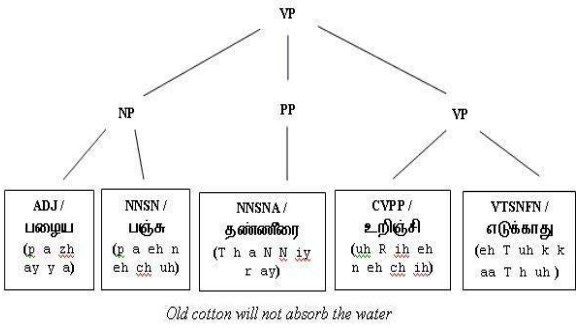

Fig. 3 A Syntax Tree

The Treebank provides training material for Machine Learning in NLP systems [19]. It is used to build gold standards for the evaluation of NLP systems. It advocates linguistic experimentation against other linguistic theories. It

provides material for human grammar exploration and learning [20].

Simple bracketed version of Treebank is generated by phrasing all the sentences in the text corpus. This is basically constituency based format. This is done manually in the initial stage. Since it is laborious and time consuming, bootstrapping is applied with manual corrections. All the annotated sentences will take form as shown below.

(S1 (S (NP (ADJ பழைய) (NNSN பஞ்சு))
(PP (NNSNA தண்ணீரை))
(VP (CVPP உறிஞ்சி)
(VTSNFN எடுக்காது))))

Transliterated Equivalent sentence

(S1 (S (NP (ADJ 'p a zh ay y a')
(NNSN 'p a eh n eh ch uh'))
(PP (NNSNA 'T h a N N iy r ay'))
(VP (CVPP 'uh R ih eh n eh ch ih')
(VTSNFN 'eh T uh k k aa T h uh'))))

D. Hybrid Model

Hybrid model is created using phrase structured Treebank. By means of immediate head parsing technique heads are selected from various constituents and trigram approach is applied among the heads. For all the parameters of constituent *c* feature files are created and updated during the training process. All the feature files together constitute the hybrid language model.

V. Experiments

A. Proposed POS Tag-set for Tamil Language

Based on rich morphological inflections and POS forms of Tamil language, more than 500 tags have been created. In Tamil Language nouns and verbs take more forms than other languages as suggested in the table 1. Preposition takes direct and noun combined forms. Adjective takes direct and verb combined forms. For interrogative statements wh-tags were generated. This POS tag set has wider coverage to all Tamil language words. Some of the examples of the tags are given in table 2.

TABLE II
Sample Tags for Tamil Language

Tag	Description	Example in Tamil with ARPABET Transliteration and meaning
ADJ	Adjective	அழகிய (a zh a k ih y a)^2 (beautiful)
ADJAP	Adjective Past participle	செய்த (ch eh y T h a)^2 (done)
ADV	Adverb	வேகமாக (v ee k a m aa k a)^2 (quickly)
CON	Conjunction	அல்லது (a l l a T h uh)^2 (or)
CVCN	Verbal Conditional negative	செய்யாவிட்டால் (ch eh y y aa v ih T T aa l)^2 (if not done)
DET	Determiner	இந்த (ih nn T h a)^2 (this)
INT	Interjection	ஐயோ (ay y oo)^2 (Alas)
NAPC	Adjective Noun plural common	நல்லவர்கள் (nn a l l a v a r k a L)^2 (good people)
ORD	Ordinal	மூன்றாவது (m uw n R aa v a T h uh)^2 (Third)
PRP	Preposition	உள்ளே (uh L L ee)^2 (inside)
QNT	Quantifier	சில (ch ih l a)^2 (few)
V	Verb	படி (p a T ih)^2 (study)
VC	Verb Causative	கற்பி (k a R p ih)^2 (teach)
VFPA	First Person Plural Past Tense Verb	சென்றோம் (ch eh n R oh m)^2 (we went)
VI	Intransitive verb	திரும்பு (T h ih r uh m p uh)^2 (turn)
VIF	Infinitive Verb	செய்ய (ch eh y y a)^2 (to do)
VSPAN	Second Person Plural Past Tense Negative Verb	செய்யவில்லை (ch eh y y a v ih l l ay y ay)^2 (did not do)
VT	Transitive Verb	திருப்பு (T h ih r uh p p uh)^2 (turn – any object)
VTSNFN	Third Person Singular Neutral Future Tense Negative Verb	எடுக்காது (eh T uh k k aa T h uh)^2 (will not take)

2 - Transliterated Tamil word using ARPABET format

B. Proposed Phrase Structures

For applying the syntactic phrases for the sentences, the following phrase notations were suggested. The proposed phrase set covers all the constituent structures of all Tamil language sentences. This is shown in table 3.

C. Generation of Phrase Structure Treebank

Phrase structure Treebank has been developed for 326 sentences which has the size of more than 5000 words by using POS tags and phrases. Initially all the 326 sentences were manually annotated and used for training. To improve the performance of the language model, additional 700 sentences are being annotated by bootstrapping followed by manual corrections.

TABLE III
Proposed Phrases

Phrases	Descriptions
NP	Noun Phrase
VP	Verb Phrase
ADVP	Adverbial Phrase
ADJP	Adjective Phrase
PP	Prepositional Phrase
CP	Conjunctional Phrase
IP	Interjectional Phrase
WHNP	Conjunctional Noun Phrase
WHVP	Conjunctional Verb Phrase
WHPP	Conjunctional Prepositional Phrase

D. Building Hybrid Language Model

Hybrid Language model has been trained with the phrase structure Treebank which comprises features files generated for the features quoted in equation (4). The probability values of all the features are initialized and updated during the training process. These values are used later in the parsing process.

By considering (*CVPP* / உறிஞ்சி ['uh R ih eh n eh ch ih']^1) as constituent *c* in figure.2, examples of the features are shown in the table 4.

TABLE IV
FEATURES IN LANGUAGE MODEL AND THEIR EXAMPLES

Features	Description	Example
T	Tag of constituent	CVPP
L	Label of the constituent	VP
H	Head of the constituent	உறிஞ்சி (uh R ih eh n eh ch ih)[3]
E	Expansion of constituent	---
M	Label of the parent	VP
I	Head of the parent	எடுக்காது (eh T uh k k aa T h uh)[3]
U	head-part-of-speech for the parent	VTSNFN

3 - Transliterated Tamil word using ARPABET format

VI. RESULTS AND DISCUSSIONS

In our experimentation, trials have been conducted with different test cases on trigram language model and phrase structured hybrid language model. Two test cases with 120 and 40 sentences have been selected from trained set and test set respectively. The results are shown in tables 5, 6 and 7. For this small vocabulary size training set, trigram model performs better than grammar model.

TABLE V
RESULTS FROM TRIGRAM MODEL

Details	Trained Sentences	Test Sentences
Total sentences	120	40
Perplexity	16.62	29.12
Entropy	4.05 bits	4.86 bits
Computation based on	943 words.	298 words
Number of 3-grams hit	911 (96.61%)	190 (63.76%)
Number of 2-grams hit	16 (1.70%)	50 (16.78%)
Number of 1-grams hit	16 (1.70%)	58 (19.46%)
Out Of Vocabularies	10	13
Context Cues	0	0

TABLE VI
RESULTS FROM PHRASE STRUCTURE HYBRID MODEL

Details	Trained Sentences	Test Sentences
Total Sentences	120	40
Correct Sentences	60	17
Sentence Accuracy	50.0%	42.5%
Ref. Words	716	228
Hyp. Words	671	221
Total Word Accuracy	94 % (671 / 716)	93% (211/228)

TABLE VII
COMPARISON RESULTS OF GRAMMAR, TRIGRAM AND HYBRID MODELS IN LOG PROBABILITIES

Grammar Model	Tri-gram Model	Hybrid Model
-19.8222	-3.55271e-15	-1.5864
-20.004	0	-1.5864
-29.2528	-3.55271e-15	-1.58641
-111.729	-39.8631	-41.4495
-97.7369	-19.9316	-21.518

Due to some reasons hybrid language model is outperformed slightly by trigram base line model. Since training set size is small, hybrid language model performs marginally lower than trigram model and highly greater than pure grammar based model but it incorporates syntax with semantics, long term relationship and free word order. The hybrid language model and the trigram model capture different facts about the distribution of words in the language, and for some set of sentences one distribution will perform better than the other.

VII. CONCLUSION AND FUTURE WORK

Hybrid language model has been built successfully which covers more than 5000 words and used for lexicalized and statistical parsing and tested with different test cases. Hybrid model performs well for the application of syntax with semantics, long term relationship and free word order. In the process of building hybrid model, suitable POS tag set for Tamil language with more than 500 tags has been generated and a phrase structure Treebank with 326 sentences has been developed. This work is being extended for additional 700 sentences to improve the performance further. In future this hybrid language model will be developed with more than 3000 sentences and dependency structured language model will be built to have the functional relationship among the words in the sentences which will lead to the best performance in the application of long term relationship and free word order.

ACKNOWLEDGMENT

The authors would like to thank Central Institute of Indian Languages (CIIL), Mysore, India and Department of Science and Technology, New Delhi, India for providing the Tamil text corpora.

ENDNOTE

1 – Transliterated Tamil words using ARPABET format

REFERENCES

[1] Stolcke, A. and Segal, J. Precise Ngram Probabilities from Stochastic Context-Free Grammars. In Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, 1994, 74–79.
[2] Chi, Z. and Geman, S, Estimation of Probabilistic Context-Free Grammars. Computational Linguistics 24 2, 1998, 299–306.
[3] Roark B. Probabilistic Top–Down Parsing and Language Modeling, Association for Computational Linguist, 2001
[4] Collins, M. J. Three Generative Lexicalized Models for Statistical Parsing. In Proceedings of the 35th Annual Meeting Of The Acl., 16–23., 1997
[5] Daniel M. Bikel, On the Parameter Space of Generative Lexicalized Statistical Parsing Models, Ph.D. Thesis, University Of Pennsylvania, 2004
[6] Daniel Jurafsky & James H. Martin, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, 2nd Edition, Pearson Education, 2006
[7] Chelba, C. And Jelinek, F. Exploiting Syntactic Structure for Language Modeling. In Proceedings for COLING-ACL 98. ACL, Newbrunswick NJ, 1998, 225–231.
[8] Collins, M. J. Head-Driven Statistical Models for Natural Language Parsing. University of Pennsylvania, Ph.D. Dissertation, 1999
[9] Brian Roark Eugene Charniak, Measuring Efficiency in High-Accuracy, Broad-Coverage Statistical Parsing Proceedings of the COLING 2000 Workshop on Efficiency in Large-Scale Parsing Systems, 2001, Pages 29-36
[10] Chelba, C. And Jelinek, F. Structured Language Modeling. Computer Speech and Language 14, 2000, 283–332.
[11] Peng Xu, Ciprian Chelba, Richer Syntactic Dependencies for Structured Language Modeling Computational Linguistics (ACL), Philadelphia, Proceedings of the 40th Annual Meeting of the Association, 2002

[12] Diego Linares Pontificia and Jos E-Miguel Benedi And Joan-Andreu Sanchez, A Hybrid Language Model based on a Combination of N-Grams and Stochastic Context-Free Grammars , ACM Transactions on Asian Language Information Processing, Volume 3, Issue 2, 2004, Pp.113-127.
[13] Ratnaparkhi, A. Learning to parse Natural Language with Maximum Entropy Models. Machine Learning 34 1/2/3, 1999, 151–176.
[14] Charniak, E. A Maximum-Entropy Inspired Parser. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics . ACL, New Brunswick NJ, 2000
[15] Eugene Charniak, Immediate-Head Parsing for Language Models, Proceeding of ACL, 2001
[16] Bharati, Akshar, Vineet Chaitanya and Rajeev Sangal, Natural Language Processing: A Paninian Perspective, Prentice-Hall of India, New Delhi, 1995
[17] Rajendran S, Strategies In The Formation Of Compound Nouns In Tamil, Languages Of India, Volume 4, 2004
[18] Marcus, M. P., Santorini, B. And Marcinkiewicz, M. A, Building A Large Annotated Corpus of English: The Penn Treebank. Computational Linguistics 19, 1993, 313–330
[19] Charniak, E. Tree-Bank Grammars. In Proceedings of the Thirteenth National Conference on Artificial Intelligence. AAAI Press/MIT Press, Menlo Park, 1996, 1031–1036.
[20] Akshar Bharati, Rajeev Sangal, Vineet Chaitanya , Anncorra : Building Tree-Banks in Indian Languages, COLING 2002 Post Conference Workshops - Proceedings of the 3rd Workshop on Asia Language Resources and International Standardization at Taipei, Taiwan, 2002.

M. Selvam received the B.E. degree in Computer Science and Engineering from Bharathidasan University, Trichy in 1990 and the M.E. Degree in Computer Science and Engineering from Bharathiar University, Coimbatore in 2002. He is currently pursuing the Ph.D., Degree at Anna University, Chennai. He is currently working as Assistant Professor in department of Information Technology at Kongu Engineering College, Perundurai, Erode. He is working in number of projects on speech recognition and synthesis, statistical natural language processing and machine translation at Speech and Language Processing Lab at Kongu Engineering College. He is also a team member of research project sponsored by Tamil Virtual University, Chennai. His areas of interest are Speech and Language Processing, Computational Linguistics, Machine Translation and Enterprise Computing.

A. M. Natarajan is Professor and Head of Computer Science and Engineering (PG) and Principal of Kongu Engineering College, Perundurai. He obtained Ph.D in Systems Engineering from P.S.G College of Technology, Coimbatore in 1984. He has published more than 100 papers in national and international journals and conferences. He was awarded "The Best Engineering College Principal Award" for the year 2000 by ISTE, New Delhi. He is member of various scientific and professional societies. He has guided more than 75 M.E and M.C.A students. Presently he is guiding many Ph.D and M.Phil students. His research areas include Software Engineering, Soft Computing, Operating Systems, Software Project Management and Networking.

R. Thangarajan received the B.E. degree in Electronics and Communication Engineering from Bharathiar University, Coimbatore in 1991 and the M.E. Degree in Computer Science and Engineering from Bharathiar University, Coimbatore in 1995. He is currently pursuing the Ph.D., Degree at Anna University, Chennai. He is currently working as Assistant Professor in department of Information Technology at Kongu Engineering College, Perundurai, Erode, India. He is working in number of projects on speech recognition and synthesis, statistical language modeling and machine translation at Speech and Language Processing Lab at Kongu Engineering College. He is also a team member of research project sponsored by Tamil Virtual University, Chennai. His areas of interest are Speech and Language Processing, Pattern Recognition, Machine Translation and Neural Networks.