

A SURVEY OF GRAMMAR CHECKERS FOR NATURAL LANGUAGES

Nivedita S. Bhirud¹, R.P. Bhavsar² and B.V. Pawar³

¹Department of Computer Engineering,
Vishwakarma Institute of Information Technology, Pune, India
^{2,3}School of Computer Sciences, North Maharashtra University, Jalgaon, India

ABSTRACT

Natural Language processing is an interdisciplinary branch of linguistic and computer science studied under the Artificial Intelligence (AI) that gave birth to an allied area called 'Computational Linguistic' which focuses on processing of natural languages on computational devices. A natural language consists of a large number of sentences which are linguistic units involving one or more words linked together in accordance with a set of predefined rules called grammar. Grammar checking is the task of validating sentences syntactically and is a prominent tool within language engineering. Our review draws on the recent development of various grammar checkers to look at past, present and the future in a new light. Our review covers grammar checkers of many languages with the aim of seeking their approaches, methodologies for developing new tool and system as a whole. The survey concludes with the discussion of various features included in existing grammar checkers of foreign languages as well as a few Indian Languages.

KEYWORDS

Natural Language Processing; Computational Linguistic; Grammar Formalism; Grammar Checker

1. INTRODUCTION

Language is a means of communication, particularly in human beings, that has origin in natural thoughts regardless of any planning or motivation. Language is subjective, creative, and dynamic system which encompasses vocal symbols of human being that play a crucial role in social affairs. Human natural language can be broadly defined as an interchangeability process within participants. It is contrasted to any natural communication system example includes bees' waggle dance or artificial/constructed language such as computer programming languages.

In today's era of technology, language engineering focuses on modeling of human languages under Computational Linguistic (CL) research domain. Computational Linguistic is an interdisciplinary field of computer science and linguistics has collaboration with Artificial Intelligence area, and is concerned with computational aspects of human natural language.

Computational linguistics is categorized into applied and theoretical components. Theoretical linguistic deals with linguistic knowledge needed for generation and understanding of language. Applied computational linguistic has concern with the development of tools, technologies, and applications to model human language. Although existing technologies are far from attaining human abilities and have major feats in related application development, computers are not able to correspond to human thoughts. As people share knowledge, ideas, thoughts and information with each other using natural language, it is also possible to share the same with computer with the help of applied CL that too, in natural languages only. To complete communication and to make it meaningful, used language must follow set of rules involved in it.

Grammar is the study of significant elements in language and set of rules that make it coherent. Words are grammatical basic units that combine together to form a sentence and collection of sentences completes the language. It is a bit easier for human beings to follow rules of native natural language as they are aware of it since infant phase. But it is a new and exciting challenge for language technology & applied CL to validate grammatical correctness of any natural language for computers. To deal with grammatical mistakes by human is also one of the challenging tasks. Even though grammar checker tools have been developed so far for many worldwide languages, it is relatively new in Indian languages. So there is scope to develop grammar checker for Indian languages.

The remaining of the paper is organized as follows: Section 2 gives idea about basic building block of language i.e. sentence and its types. Section 3 outlines fundamental concepts of grammar checking system. Section 4 gives general working of grammar checker. Section 5 describes available grammar checking approaches. Section 6 discusses how couple of world-wide natural languages, including few Indian natural languages use grammar checking approach. After descriptive study, analysis is shown in section 7. The review is summary and observations are recorded in section 8.

2. A BIT ABOUT LANGUAGE

Language is a mean of communication. Human beings exchange information between two or more parties using natural languages only. The prime objective of communication is to share information and request/impart knowledge. The information can be specified in *written-form* or *vocal-form*(spoken). The most important thing in information content form is the validity of sentences in the given language. Morphemes, phonemes, words, phrases, clauses, sentences, vocabulary and grammar are the building blocks of any natural language. All valid sentences of a language must follow the rules of that language (grammar). Invalid sentences are not worth and won't be effective to share knowledge, hence outrightly rejected.

Any natural language consists of countably infinite sentences and these sentences follow basic structure. A sentence structure is perceived hierarchically at different levels of abstraction, i.e. surface level(at the word level), POS(part-of-speech) level to abstract level(phrases: subject, object, verb etc.).The sentence formation strictly depends on the syntactically permissible structures coded in the language grammar rules. The basic sentence structures broadly depend on the positions of Subject, Object, Verb i.e. their permutations, accordingly SVO, SOV, OSV, OVS, VSO, VOS are possible, but not all(OVS, OSV) are followed in the grammar of natural languages of the world. These are all referred as word order. Depending the internal phrasal

structure of phrases especially the verb phrases, certain clauses, sentences are broadly classified as Simple, Complex and Compound sentence.

2.1 Simple Sentence

A simple sentence is a collection of predicate and one or more arguments. It has only single main clause and single verb (mostly verb root). It expresses a meaning that can stand by on its own. It does not contain any negation, question words and passivation. It has simpler sentence structure. Most of these sentences are kernel sentences and can span other sentence types such as complex and compound sentences. It is identified using morphology of the verb phrase, number of clauses and other cue words are negation words, question words and passivation.

2.2 Complex Sentence

A complex sentence has at least two clauses, having interdependence between main and dependent or subordinate clause. Subordinate clause gives additional information about main clause. Number of clauses is the important cue for identifying the Complex Sentences.

2.3 Compound Sentence

A compound sentence is a collection of multiple clauses connected through conjunctives (And/Or etc.), here it is important to note that each clause is a functionally complete sentence (having complete meaning). As there is no interdependency between clauses, they are considered at the same (peer) level. The use of conjunctive words is important cue for identifying this type of sentences.

3. GRAMMAR CHECKER

Now a days, people need not only mechanical support, but also expect intellectual assistance from machines. What if, we could have conveyed everything in natural languages to machines? Answers to such questions open new doors of possibilities and opportunities for intelligent systems. For realizing this idea into reality, it is a mandatory criterion for the machines to get aware of natural languages. It motivates us to build intelligent computer applications involving scientific and linguistic knowledge of human communication. Applications of NLP include QA (question-answering) system, Machine Translation, NL (Natural language) interface to databases, grammar checker, spell checker, chatter box, etc. Grammar checking is the fundamental application amongst these as it checks correctness of input sentence, which has a strong effect on other NLP applications. Correctness and validity of sentences are checked with the help of an underlying grammar of the natural language. The grammar consists of a set of rules which govern the formation and amalgamation of sentence constituents i.e. clauses/phrases. A valid phrase is one in which all constituent words are compatible with each other so to say they satisfy the morphological/syntactic agreement features, i.e. they agree in their *gender, number, person, case* (GNPC) features. The phrase structure is defined with the help of POS (part-of-speech) sequence. The same is true for a sentence in which the main verb agrees with either *subject, object* or none. Grammar checking involves testing the agreements between constituents on the scale of GNPC syntactic features as well as ontological semantic features.

At practical level, it is highly desired that the grammar checker software should not only check the correctness and validity of the sentences, but also correct the grammatical errors (auto correction, correction suggestion). The word order(kernel sentence structure) affects the parsing of the given natural language so does the NLP tasks and tools used for the grammar checking process.

4. GENERAL WORKING OF GRAMMAR CHECKER

Grammar checker takes as input, a sentence from the given document of a language and checks its correctness and validity w.r.t that language [1]. The sentence has to undergo some kind of preprocessing stage where in lexicalization, word setgenation, POS tagging. The actual grammar checking involves syntactic parsing of pre-processed sentence using chosen approach (discussed below) . The output of the Grammar Checker application should summarize the grammatical errors in the input document sentences and optionally auto correct the errors or at least provide correction suggestions. These tasks can be described in algorithmic form(step wise) as follows:

1. **Sentence Tokenization:** This involves sentence tokenization and word segmentation. The sentence is tokenized into words followed by breaking down words into constituent morphs and populating lexical information about the word from the lexicon.
2. **Morphological Analysis:** Morphological analysis returns word stems and associated affixes.
3. **Part-of-Speech(POS) tagging:** Assigning the appropriate POS tag to each word (morpheme)
4. **Parsing Stage:** Checks the syntactic constraints (agreement constraints) between input words and formation of Hierarchical phrasal/dependency structure of the input sentence using chosen approach/methodology. In case of failure flag grammatical error also provide auto correction mechanism or present suggestion list to the user.

Many grammar checkers have been developed for different foreign and Indian languages using different approaches. A detailed description is given in section 5.

5. GRAMMAR CHECKING APPROACHES

Broadly, there are three grammar checking approaches that are used, namely statistical, rule based and hybrid grammar checker.

- A) Statistical Grammar Checker
- B) Rule based grammar Checker
- C) Hybrid Grammar Checker

5.1 Statistical Grammar Checker

In this approach, corpus is maintained from a number of journals, magazines or documents. It ensures correctness of sentence by checking the input text against the corpus. There are two ways to check input text. First, input text is directly checked with corpus and matched input text, is tagged as grammatically correct otherwise it is considered as incorrect. In the second way, initially rules are generated from maintained corpus and input text is checked by these rules. But rules need to be updated when the corpus is modified or new data is added to it. It is easy to implement, but has some disadvantages also. As there is no specific error message, it is difficult to recognize error given by a system [3].

5.2 Rule based Grammar Checking

It is the most common approach. Input text is checked by rules formed from corpus, but unlike the statistical approach, rules are generated manually [12]. However, rules are easy to configure and also easy to add, remove or update. The most significant advantages include, rules can be handled by one who do not have programming language like linguistics and it provides a detailed error message. The main feature of this method is that it provides all feature of a language and sentences also need not to be complete, it can handle input text while writing.

5.3 Hybrid Grammar Checker

It combines both statistical and rule based grammar checker. So it is more robust and achieves higher efficiency.

6. EXISTING GRAMMAR CHECKER

This section will provide a detailed study of the existing grammar checker for world-wide languages. For study purposes, languages are divided into categories; foreign and Indian.

6.1 Grammar Checker for Foreign Languages

6.1.1 Afan Oromo Grammar Checker

Afan Oromo is the language of Ethiopia. The Afan Oromo grammar checker[2] is provided with paragraphs as an input, which is then, tokenized into sentences, further into words. Using tagger based Hidden Markov Model, which uses manually tagged corpus, each word is assigned a part of speech. A stemming algorithm removes certain types of affix using substitution rules, which only apply when certain conditions hold. By removing affix, an agreement between a subject - verb, subject-adjective, main verb-subordinate verb in number, gender, and the tense is identified. From identifying agreements, the system can provide alternative correct sentences in case of disagreement. The grammar checker provides prominent results but fails to detect compound and complex grammatically incorrect sentences. Also, as system is provided with incorrectly tagged words, it leads the system to generate false flags [2].

6.1.2 Amharic Grammar Checker

Speakers of Amharic language are found in Ethiopia. The Amharic grammar checker system [3] adopts two approaches. The first approach is a rule based approach for simple sentences. Another approach is a statistical approach for both simple and complex sentences. The rules are created manually and checked against the pattern of a sentence to be checked. To check grammatical errors in an Amharic sentence, n-gram and probabilistic methods are used along with a statistical approach. The pattern and the corresponding occurrence probabilities are extracted automatically from training corpus and maintained in the repository. Using stored patterns and probabilities, sentence probability can be calculated. Some threshold and probability of the sentence are used to determine correctness. The system shows good results, but false alarm is due to incomplete grammatical rules and quality of the corpus [3].

6.1.3 Swedish Grammatifx Grammar Checker

The speakers of Swedish are found in Sweden and parts of Finland. Grammatifx is an exploratory project, which has been developed as a Swedish grammar checker [4]. The study starts with investigating existing grammar checker of different languages. Swedish materials are collected for gathering error types and for the discovery of new ones. Error type in this classification is evaluated and subset of these error types is chosen for actual project development. Unlike other Swedish grammar checker [8], Grammatifx checks noun phrase internal agreement and verb chain consistency. To detect various error types, different technologies are selected. For detection of syntactic errors, constraint grammar formalism is used. Regular expression based technique is used to detect punctuation and number formatting convention violations. Word-specific stylistic marking is covered by style-tagging individual lexeme entries in the underlying Swedish two-level lexicon. Along with error detection, Grammatifx also has an error treatment module. It does not detect compound words mistakenly written separately.

6.1.4 Icelandic Grammar Checker

Icelandic is a language of Iceland. Rule based approach is used to implement Icelandic grammar checker [5]. Initially, input text has to pass through parsing, for syntactic analysis and POS tagging. After initial analysis of input text, the system then goes through part of finding process relevant to each rule. The system finds compliance with the rule. If it finds that in some way, input text is not in accordance with relevant rules, an error is generated. This system does only general error detection; it does not provide any detailed error message or correction suggestions. It also does not detect stylistic errors.

6.1.5 Nepali Grammar Checker

Nepali language is an official language of Nepal. Bal Krishna Bal and Prajol Shrestha have presented works on a Nepali grammar checker using rule based approach [6]. Input text is tokenized into words, then the word is passed to the morphological module for initial POS tagging. The next module is a POS tagger module, which tags untagged and undetermined tokenized words. Chunker and parser module identifies the chunks or phrases from POS tagged words. Thus, this module requires production rules and POS tags of input texts and after matching with the rule, it will return chunks and phrases. Identified chunks, and phrases are

assigned to grammatical roles like SUBJECT, OBJECT and VERB based on subject-verb agreement and agreement between modifier and head in the noun phrase. The syntax is checked then. The grammar checker only deals with simple sentences.

6.1.6 Portuguese Grammar Checker

Portuguese is a Romance language. Portuguese grammar checker named COGrOO is based on CETENFOLHA, a Brazilian Portuguese morphosyntactic annotated Corpus [7]. The system consists of local and structural error rules. Local rules include short word sequence rules, whereas structural rules include nominal-verbal agreement, nominal-verbal government, and misuse of adverb-adjective type complex rules. Initially, inputs are broken down into sentences by boundary detector module. Then each sentence is tokenized into words, POS tags are assigned to each of them. Chunker will chunk a tagged sentence into small verbal and a noun phrase with their respective grammatical roles. Local errors are detected by the local error checker whereas structural errors are detected by structural error checker.

6.2 Grammar Checker for Indian Languages

6.2.1 Urdu Grammar Checker

Kabir proposed the Urdu grammar checker system. The proposed two pass parsing approach analyzes the input text [15]. This approach was introduced to reduce redundancy in phrase structure, grammar rules developed for sentence analysis. Phrase structure grammar rules are used to parse the sentence. In case of failure, Movement rules are used to reparse the tree. The system works well, except for the module of Morphological disambiguation and POS guesser. The grammar checker checks grammatical and structural mistakes in declarative sentences and provide error correction suggestions.

6.2.2 Bangla Grammar Checker

Md. Jahangir Alam, Naushad UzZaman and Mumit Khan describes n gram based analysis of words and POS tags to decide the correctness of a sentence [9]. The system assigns tags to each word of a sentence using a POS tagger. Using-gram analysis, the system determines probability of tag sequence. If the probability is greater than zero, then it considers the sequence as correct. The same system is tested for both English and Bangla language. As it completely depends on POS tagging, author checked system against manual tagged sentences and automated tagged and results are more promising for Bangla compared to English, as there are large compound sentences in Brown corpus. The system also does not work for compound sentences of Bangla language.

6.2.3 Punjabi Grammar Checker

Mandeep Gill, Gurprit Lehal and Shiv Sharma Joshi have implemented grammar checking software for detecting grammatical errors in Punjabi texts and have provided suggestions [10]. The system performs morphological analysis using a full form lexicon and POS tagging and phrase chunking using rule based system. For literacy style Punjabi texts, the system also supports a set of carefully devised error detection rules, which can detect alteration for various

grammatical errors generated from each of the agreements, order of words in phrases etc. The rules in the system can be turned off or on individually. The main attraction of this system is that it provides a detailed description of detected errors and provides suggestions on the same. It also deals with compound and complex sentences.

6.2.4 Hindi Grammar Checker

Lata Bopche, Gauri Dhopavkar and Manali Kshirsagar describes a method for Hindi grammar checker [17]. This system consists a full-form lexicon for morphological analysis and rule based system. The input text passed through all basic processes like tokenization, morphological analysis, POS tagging. A POS tagged sentences match against a set of rules. This system gives prominent results for only simple sentences. The system only checks those patterns which have the same number of words present in the input sentence. It does not provide any suggestion for the errors

6.3 Commercial Grammar Checkers

A lot of work has gone into developing commercially sophisticated systems for widespread use, such as automatic translator, spell checkers, grammar checkers and so on for natural languages. However, most of such programs are available strictly on commercial basis, therefore no official documentation regarding their approaches/algorithms is available. Such commercial programs are available on Proprietary as well as Open Source office suite software such MS Microsoft word, open office, Libre office suite and many more. Majority of such grammar checkers are available for English language.

7. ANALYSIS

After descriptive study of various grammar checks for world-wide languages, some findings are analyzed. Table 1 summarizes language and respective grammar checker approach with performing and lacking features in each of them.

Table 1. Summarization of grammar checker features

Language	Feature of language	Grammar checker approach	Performing Features	Lacking Features
Afan Oromo[2]	<ul style="list-style-type: none"> Rich in morphology Agglutinative language 	Rule based	Provides alternative sentence in case of disagreement	Fails to detect compound and complex grammatically incorrect sentence Poor POS tagger
Amhari [3]	<ul style="list-style-type: none"> Rich in morphology Subject-Object-Verb structure 	Hybrid	Results are prominent in both simple and complex sentences. Able to detect multiple errors in a sentence	Gives false alarm due to incomplete rules and quality of statistical data

Swedish [4]	<ul style="list-style-type: none"> High amount of word independence 	Hybrid	Error detection and error correction module available Detects syntactic, stylistic and punctuation error	Does not detect compound words mistakenly written separately.
Icelandic [5]	<ul style="list-style-type: none"> Heavily inflected 	Rule based	Able to detect general errors	Does not provide detailed messages of error and also not correct it.
Nepali[6]	<ul style="list-style-type: none"> Family of Hindi and Bangala language 	Modular	Only simple sentences are checked and error messages are provided	Does not deal with compound and complex sentence
Portuguese [7]	<ul style="list-style-type: none"> Infinitive language 	Rule based	Local and structural errors detected	Lacks detection of stylistic error
Urdu[15]	<ul style="list-style-type: none"> Subject-Object-Verb structure 	Rule based	Checks structural and grammatical mistakes in declarative sentences. Provides error correction.	Weak in performance by morphological disambiguation and POS tagger
Bangla[9]	<ul style="list-style-type: none"> Loaded with the agreement 	Statistical	The system gives better results for Bangla language compared with English	Does not work for compound sentence.
Punjabi[10]	<ul style="list-style-type: none"> Concern with word order, case making, verb conjugation 	Hybrid	Provides detailed message of the error and give error correction suggestion. Works for simple, compound and complex sentences.	Due to word shuffling for emphasis, false alarm occurs. Emphatic intonation causes changes in meaning or class of the word.
Hindi[17]	<ul style="list-style-type: none"> Word dependency Inflectional rich 	Rule based	Gives promising results for simple sentences	The system only checks those patterns which have the same number of words present in the input sentence.

Table 2 summarizes error detection and/or correction, working on simple and/or compound and/or complex sentences, syntactic and stylistic error.

Table 2. Summarization of Detected errors by grammar checker

Language	Error Detection	Error Correction	Detect Error for			Syntactic Error	Stylistic Error
			Simple Sentence	Compound Sentence	Complex Sentence		
Afan Oromo[2]	✓		✓	✓	✓		
Amharic [3]	✓		✓		✓		
Swedish[4]	✓	✓	✓		✓	✓	
Icelandic[5]	✓		✓		✓		
Nepali[6]	✓		✓		✓		
Portuguese[7]	✓		✓		✓		

Urdu[15]	✓	✓	✓			✓	
Bangla[9]	✓		✓			✓	
Punjabi[10]	✓	✓	✓	✓	✓	✓	
Hindi[17]	✓		✓			✓	

Table 3 provides performance evaluation of studied grammar checkers.

Table 3. Performance evaluation of grammar checker

Language	Grammar checker approach	Performance Evaluation
Afan Oromo[2]	Rule based	Precision: 88.89% Recall: 80.00%
Amharic [3]	Rule Based	Precision: 92.45 % Recall: 94.23%
	Statistical	Precision: 67.14% Recall: 90.38%
Swedish[4]	Hybrid	Not reported
Icelandic[5]	Rule based	Precision: 84.21% Recall: 71.64%
Nepali[6]	Rule Modular	Not reported
Portuguese[7]	Rule based	A corpus used contains 51 errors based on which following parameters calculated True Positive: 14 & False Positive: 10
Urdu[15]	Rule based	Not reported
Bangla[9]	Statistical	Calculate probability of tag sequence and if it is above the threshold value, it is concluded as probabilistically correct otherwise not. Few examples are provided.
Punjabi[10]	Hybrid	Not reported
Hindi[17]	Rule based	Not reported

8. CONCLUSION

In this survey, we have reviewed different grammar checking approaches, methodologies along with key concepts and grammar checker internals. The aim of the survey was to study various Grammar Checkers on the scale of their features such as types of grammar errors, weaknesses and evaluation. Survey concludes with study of various features of grammar checkers thus leading to future scope for developing grammar checkers for uncovered languages with feasible approach. It is observed that most of the professionally available grammar checkers are available for English language, while for most of other languages, the work is in progress (early to final stages). Number of Indian Initiatives were also witnessed during our survey for popular languages like Hindi, Bangla, Urdu etc. No grammar checker could have been cited for Marathi Language. Hence our future research work aims to develop grammar checker for Marathi Language.

REFERENCES

- [1] Misha Mittal, Dinesh Kumar, Sanjeev Kumar Sharma, “Grammar Checker for Asian Languages: A Survey”, International Journal of Computer Applications & Information Technology Vol. 9, Issue I, 2016
- [2] DebelaTesfaye, “A rule-based Afan Oromo Grammar Checker”, International Journal of Advanced Computer Science and Applications, Vol. 2, No. 8, 2011
- [3] Aynadis Temesgen Gebru, ‘Design and development of Amharic Grammar Checker’, 2013
- [4] Arppe, Antti. “Developing a Grammar Checker for Swedish”. The 12th Nordic conference computational linguistic. 2000. PP. 13 – 27.
- [5] “A prototype of a grammar checker for Icelandic”, available at www.ru.is/~hrafm/students/BScThesis_Prototype_Icelandic_GrammarChecker.pdf
- [6] Bal Krishna Bal, Prajol Shrestha, “Architectural and System Design of the Nepali Grammar Checker”, www.pan110n.net/english/.../Nepal/Microsoft%20Word%20-%208_OK_N_400.pdf
- [7] Kinoshita, Jorge; Nascimento, LaHS do; Dantas ,Carlos Eduardo. ”CoGrOO: a Brazilian-Portuguese Grammar Checker based on the CETENFOLHA Corpus”. Universidade da Sro Paulo (USP), Escola Politřcnica. 2003.
- [8] Domeij, Rickard; Knutsson, Ola; Carlberger, Johan; Kann, Viggo. “Granska: An efficient hybrid system for Swedish grammar checking”. Proceedings of the 12th Nordic conference in computational linguistic, Nodalida- 99. 2000.
- [9] Jahangir Md; Uzzaman, Naushad; Khan, Mumit. “N-Gram Based Statistical Grammar Checker For Bangla And English”. Center for Research On Bangla Language Processing. Bangladesh, 2006.
- [10] Singh, Mandeep; Singh, Gurpreet; Sharma, Shiv. “A Punjabi Grammar Checker”. Punjabi University. 2nd international conference of computational linguistics: Demonstration paper. 2008. pp. 149 – 132.
- [11] Steve Richardson. “Microsoft Natuaral language Understanding System and Grammar checker”. Microsoft.USA, 1997.
- [12] Daniel Naber. “A Rule-Based Style And Grammar Checker”. Diplomarbeit. Technische Fakultät Bielefeld, 2003
- [13] “Brief History of Grammar Check Software”, available at: <http://www.grammarcheck.net/brief-history-of-grammar-check-software/>, Accessed On October 28, 2011
- [14] Gelbukh, Alexander. “Special issue: Natural Language Processing and its Applications”. InstitutoPolitécnico Nacional. Centro de InvestigaciónenComputación. México 2010.
- [15] H. Kabir, S. Nayyer, J. Zaman, and S. Hussain, “Two Pass Parsing Implementation for an Urdu Grammar Checker.”
- [16] Jaspreet Kaur, Kamaldeep Garg, “ Hybrid Approach for Spell Checker and Grammar Checker for Punjabi,” vol. 4, no. 6, pp. 62–67, 2014.

- [17] LataBopche, GauriDhopavkar, and ManaliKshirsagar, "Grammar Checking System Using Rule Based Morphological Process for an Indian Language", Global Trends in Information Systems and Software Applications, 4th International Conference, ObCom 2011 Vellore, TN, India, December 9-11, 2011.
- [18] MadhaviVaralwar, Nixon Patel. "Characteristics of Indian Languages" available at "http://www.w3.org/2006/10/SSML/papers/CHARACTERISTICS_OF_INDIAN_LANGUAGES.pdf" on 30/12/2013
- [19] Kenneth W. Church and Lisa F. Rau, "Commercial Applications of Natural Language Processing", COMMUNICATIONS OF THE ACM, Vol. 38, No. 11, November 1995
- [20] ChandhanaSurabhi.M, "Natural Language Processing Future", Proceedings of International Conference on Optical Imaging Sensor and Security, Coimbatore, Tamil Nadu, India, July 2-3, 2013
- [21] ER-QING XU, "NATURAL LANGUAGE GENERATION OF NEGATIVE SENTENCES IN THE MINIMALIST PARADIGM", Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2005
- [22] Jurafsky Daniel, H., James. Speech and Language Processing: "An introduction to natural language processing, computational linguistics and speech recognition" June 25, 2007.
- [23] Aronoff, Mark; Fudeman, Kirsten. "What is Morphology?". Blackwell publishing. Vol 8. 2001.
- [24] S., Philip; M.W., David. "A Statistical Grammar Checker". Department of Computer Science. Flinders University of South Australia. South Australia, 1996
- [25] Mochamad Vicky Ghani Aziz, Ary Setijadi Prihatmanto, Diotra Henriyan, Rifki Wijaya, "Design and Implementation of Natural Language Processing with Syntax and Semantic Analysis for Extract Traffic Conditions from Social Media Data", IEEE 5th International Conference on System Engineering and Technology, Aug. 10-11, UiTM, Shah Alam, Malaysia, 2015
- [26] Chandhana Surabhi.M, "Natural Language Processing Future", Proceedings of International Conference on Optical Imaging Sensor and Security, Coimbatore, Tamil Nadu, India, July 2-3, 2013