GRAMMAR CHECKERS FOR NATURAL LANGUAGES: A REVIEW

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

1 Department 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 Linguistics’ which focuses on processing of natural languages on computational devices. A natural language consists of many sentences which are meaningful linguistic units involving one or more words linked together in accordance with a set of predefined rules called ‘grammar’. Grammar checking is fundamental task in the formal world that validates sentences syntactically as well as semantically. Grammar Checker tool is a prominent tool within language engineering. Our review draws on the till date development of various Natural Language grammar checkers to look at past, present and the future in the present context. Our review covers common grammatical errors, overview of grammar checking process, grammar checkers of various languages with the aim of seeking their approaches, methodologies and performance evaluation, which would be great help for developing new tool and system as a whole. The survey concludes with the discussion of different features included in existing grammar checkers of foreign languages as well as a few Indian Languages.

KEYWORDS

Natural Language Processing, Computational Linguistics, Writing errors, Grammatical mistakes, 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[29]. 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[27]. 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 embark on common grammatical mistakes in sentence. Section 5 gives general working of grammar checker. Section 6 describes available grammar checking approaches. Section 7 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 8. The review is summary and observations are recorded in section 9.

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.))[29].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. COMMON GRAMMATICAL MISTAKES

Language communication takes place in verbal or written form. Syntactic as well as semantic validity (grammatical validity) of sentences is a pre-condition for effective communication. Inspite of formal/informal language training, human mind is prone to grammatical mistakes which range from casual/habitual to serious mistakes affecting the overall meaning of the sentence in verbal as well as written mode. Verbal mistakes have higher degree of tolerance over written form mistakes due to its formal/legal applications and implications of the later. During our work, we observed core reasons which trigger following commonly occurring grammatical mistakes. These reasons include: Word misplacement, missing words, superfluous words, sense misuse (in appropriate usage). We have categorized the common grammatical mistakes in to following categories and subcategories [30]:

1. Punctuational mistakes wrt punctuation markers viz. Comma, hyphen, Full stop, Many more
2. Constituents’ Agreement mistakes
   a. Noun-verb agreement
   b. Gender, number, case, person agreement
   c. Noun-adjective agreement
   d. Agreement in phrases (noun phrase, preposition phrases verb phrase)
   e. Clause level errors (Clausal constructions, agreement, linking, focus)
   f. And many more.
3.Misplaced modifier
4. Vegue pronominal reference
5. Inappropriate vocabulary choice (incorrect word sense)
6. Lack of parallel structure (diversified structures under same theme)
7. Sentence sprawl (sentence linking-semantic flow, elaboration vs. summarization)
8. Tense, Aspect, Modality agreement
9. And many more.
Mistakes is nothing but error. Checking of sentence is finding out error in sentence and this task can be accomplished by grammar checker.

4. 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.

5. General Working Of Grammar Checker

Grammar checkers accepts input in form of documents, paragraphs or sentence. However it then break down input into unit form, sentence. Corresponding language punctuation marks are used to identify completion of sentence. The sentence has to to undergo some kind of preprocessing. According to our study, generally, for checking validity of sentenc, following modules are used to process theses sentence.

These modules 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 morphemes 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(morph).

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.

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

6. Grammar Checking Approaches

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

A) Rule based Grammar Checker
B) Data driven Grammar Checker
C) Hybrid Grammar Checker

6.1 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]. It requires expertise and extensive labour. However, rules are easy to configure and also programmed[27]. 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. But it fails in the situation viz rule aggregation, rule collision and multiple rule qualification.

6.2 Data Driven Grammar Checker

This approach utilizes a corpus which may be crafted manually or extracted automatically from a number of journals, magazines or documents, online resources etc. Corpus is supposed to be complete minimum document of the language representing all language features and linguistic phenomenon of the language under consideration. Under data driven approach, two sub approaches involved. The first, corpus based approach, ensures correctness of sentence by checking the input text against the corpus. The second approach deals with probabilistic/statistical checking of input text. In statistical approach, annotate corpus is used to generate parts of speech (PoS) tags. These tag sequenceses will be very common or few will not occur at all. Correctcility occurring sequences will be considered correct and uncommon sequence will lead to error[27]. The environment can easily controlled is the advantage of this approach.[3]. But it requires relevant good corpus for efficient output.

6.3 Hybrid Grammar Checker

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

7. Existing Grammar Checker

This section will provide a detailed study of the existing grammar checker for world-wide languages. Along with grammar checker study, we studied a bit about these languages[28]. Following table will summerize the study of language.
Table 1. Summarization of languages

<table>
<thead>
<tr>
<th>Language</th>
<th>Country</th>
<th>Speakers Approximely</th>
<th>Year</th>
<th>Availability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Swedish[4]</td>
<td>Sweden and Finland, and also in Estonia, Norway, Canada, USA</td>
<td>9 million</td>
<td>1999</td>
<td>Open Source Grammatifex Software</td>
</tr>
<tr>
<td>Icelandic[5]</td>
<td>Iceland, Canada, and USA</td>
<td>300,000 speakers</td>
<td>Not available</td>
<td>Proprietary</td>
</tr>
<tr>
<td>Portuguese[7]</td>
<td>Portugal and Brazil</td>
<td>220 million</td>
<td>2003</td>
<td>Proprietary</td>
</tr>
<tr>
<td>Urdu[15]</td>
<td>Pakistan, India, Afganistan, Bahrain, Bangladesh, germane, Thailand</td>
<td>104 million</td>
<td>Not available</td>
<td>Proprietary</td>
</tr>
<tr>
<td>Punjabi[10]</td>
<td>India, Pakistan</td>
<td>130 million</td>
<td>Not available</td>
<td>Proprietary</td>
</tr>
<tr>
<td>Hindi[17]</td>
<td>India</td>
<td>545 million</td>
<td>Not available</td>
<td>Proprietary</td>
</tr>
</tbody>
</table>

For study purposes, languages are divided into categories; foreign and Indian.

7.1 Grammar Checker for Foreign Languages

7.1.1 Afan Oromo Grammar Checker

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].
7.1.2 Amharic Grammar Checker

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].

7.1.3 Swedish Grammatifix Grammar Checker

The speakers of Swedish are found in Sweden and parts of Finland. Grammatifix 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], Grammatifix 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, Grammatifix also has an error treatment module. It does not detect compound words mistakenly written separately.

7.1.4 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.

7.1.5 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.
7.1.6 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.

7.2 Grammar Checker for Indian Languages

7.2.1 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.

7.2.2 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.

7.2.3 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.
7.2.4 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.

7.3 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/algorithm is available. Such commercial programs are available on Propreitary as well as Open Source office suite software such MS Microsoft word, open office, Libre office suite, Grammarly and many more. Majority of such grammar checkers are available for English language.

8. ANALYSIS

After descriptive study of various grammar checks for world-wide languages, some findings are analyzed. Table 2 summarizes language and respective grammar checker approach with performing and lacking features in each of them.

<table>
<thead>
<tr>
<th>Language</th>
<th>Feature of language</th>
<th>Grammar checker approach</th>
<th>Performing Features</th>
<th>Lacking Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Afan Oromo[2]</td>
<td>• Rich in morphology</td>
<td>Rule based</td>
<td>Provides alternative sentence in case of disagreement</td>
<td>Fails to detect compound and complex grammatically incorrect sentence Poor POS tagger</td>
</tr>
<tr>
<td></td>
<td>• Agglutinative language</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Amharic [3]</td>
<td>• Rich in morphology</td>
<td>Hybrid</td>
<td>Results are prominent in both simple and complex sentences. Able to detect multiple errors in a sentence</td>
<td>Gives false alarm due to incomplete rules and quality of statistical data</td>
</tr>
<tr>
<td></td>
<td>• Subject-Object-Verb structure</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Language</td>
<td>Characteristics</td>
<td>Type</td>
<td>Error Detection/Correction</td>
<td>Remarks</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------------------------------------------</td>
<td>--------------</td>
<td>-----------------------------</td>
<td>----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Swedish[4]</td>
<td>High amount of word independence</td>
<td>Hybrid</td>
<td>Error detection and error correction module available&lt;br&gt;Detects syntactic, stylistic and punctuation error</td>
<td>Does not detect compound words mistakenly written separately.</td>
</tr>
<tr>
<td>Icelandic[5]</td>
<td>Heavily inflected</td>
<td>Rule based</td>
<td>Able to detect general errors</td>
<td>Does not provide detailed messages of error and also not correct it.</td>
</tr>
<tr>
<td>Nepali[6]</td>
<td>Family of Hindi and Bangala language</td>
<td>Rule based</td>
<td>Only simple sentences are checked and error messages are provided</td>
<td>Does not deal with compound and complex sentence.</td>
</tr>
<tr>
<td>Portuguese [7]</td>
<td>Infinitive language</td>
<td>Rule based</td>
<td>Local and structural errors detected</td>
<td>Lacks detection of stylistic error.</td>
</tr>
<tr>
<td>Bangla[9]</td>
<td>Loaded with the agreement</td>
<td>Statistical</td>
<td>The system gives better results for Bangla language compared with English</td>
<td>Does not work for compound sentence.</td>
</tr>
<tr>
<td>Punjabi[10]</td>
<td>Concern with word order, case making, verb conjugation</td>
<td>Rule based</td>
<td>Provides detailed message of the error and give error correction suggestion. Works for simple, compound and complex sentences.</td>
<td>Due to word shuffling for emphasis, false alarm occurs. Emphatic intonation causes changes in meaning or class of the word.</td>
</tr>
<tr>
<td>Hindi[17]</td>
<td>Word dependency&lt;br&gt;Inflectional rich</td>
<td>Rule based</td>
<td>Gives promising results for simple sentences</td>
<td>The system only checks those patterns which have the same number of words present in the input sentence.</td>
</tr>
</tbody>
</table>

Table 3 summarizes error detection and/or correction, working on simple and/or compound and/or complex sentences, syntactic and stylistic error.
### Table 3. Summarization of Detected errors by grammar checker

<table>
<thead>
<tr>
<th>Language</th>
<th>Error Detection</th>
<th>Error Correction</th>
<th>Detect Error for Simple Sentence</th>
<th>Compound Sentence</th>
<th>Complex Sentence</th>
<th>Syntactic Error</th>
<th>Stylistic Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Afan Oromo[2]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Amharic [3]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Swedish[4]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Icelandic[5]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Nepali[6]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Portuguese[7]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Urdu[15]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Bangla[9]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Punjabi[10]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Hindi[17]</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 4 provides performance evaluation of studied grammar checkers.

### Table 4. Performance evaluation of grammar checker

<table>
<thead>
<tr>
<th>Language</th>
<th>Grammar checker approach</th>
<th>Performance Evaluation</th>
</tr>
</thead>
</table>
| 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 based               | 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]    | Rule based               | Not reported             |
| Hindi[17]      | Rule based               | Not reported             |
9. 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.

First observed commonly occurring grammatical mistakes. The review started with studying understanding cog mi & reas o.

References


[29] Nitin Indurkhya & Fred J. Damerau, ”A Handbook of Natural Language Processing” Cambridge UK
