Component-Wise Scrutiny of Existing Rule-Based Punjabi Grammar System and Implication for Accuracy Determination

Vikas Verma and S. K. Sharma

Abstract The use of Computational Linguistics (CL) to process Natural Languages (NL) is an important domain of Natural Language Processing (NLP). Fewer grammar checkers are available implicitly for Indian literary languages despite listed twenty two languages as per eighth schedule of Indian Constitution. Traditionally, "linguistic units"—token of a sentence in literary context are grouped together according to a set of predefined rules which can be stated as "Grammar" and hence directs research to a Grammar checker which performs the task of detecting and correcting grammatical errors in the text. This paper categorically explores existing Rule-based Punjabi Grammar System by providing a framework for quantitatively measuring the effect of each component and thus overall implication of the grammar checker using precision and recall as parameters for accuracy criteria and digs out Morphological Analyzer and POS Tagger as the faulty components generating false-alarms and errors to the tune of 58.13% and 26.74%, respectively. Based on these detections, further research can be carried out for developing a model to overcome these ambiguities using Machine Learning techniques.

Keywords Computational linguistics · Natural language processing · Grammar checking · Punjabi grammar checker · Grammatical errors

1 Introduction

During communiqué, "Language" acts as a model for transferring information through a standardized approach called its grammar. A Grammar Checker used in arena of Machine Learning integrates an application of Artificial Intelligence with Computational Linguistic. The generalized functionality can be depicted in Fig. 1.

Though heaps of research is carried out in Grammar Check particularly for English and Foreign languages yet fewer research is carried out for various Indian languages like Punjabi. Statistics reveal that there are 6,900 spoken languages throughout the world. Punjabi language falls under the top ten languages with 120 million total

V. Verma (✉) · S. K. Sharma
CSA Department, D A V University, Jalandhar 144001, India

P. K. Mallick et al. (eds.), *Electronic Systems and Intelligent Computing*, Lecture Notes in Electrical Engineering 860, https://doi.org/10.1007/978-981-16-9488-2_20

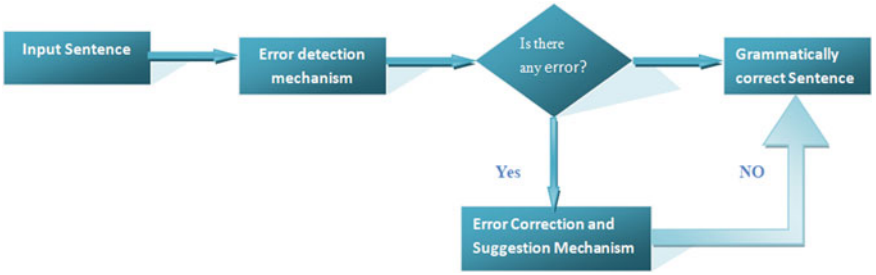

Fig. 1 Diagram for grammar checker—functionality

speakers out of which 109 million are native speakers whereas Mandarin is the top spoken language and English occupies fourth rank in the list. The tyranny of the situation is that on the internet, English reserves the lion's share of 26.8%, Chinese occupies 24.2%, Spanish maintains 7.8%, whereas all other languages contribute meager 26.4%. This drift is sufficient motivation for the research community to contribute in this sphere. Also, the Punjabi language finds its linkage to Indo–Aryan languages family generally referred to as Indic Languages and is morphologically rich language.

Grammar checking systems are mostly an integral part of specific word processors. For instance, in English language, by default characteristic is imbibed in Microsoft Office and for Punjabi, such functionality is provided in AKHAR (a software exclusively designed for literary purpose). Contribution in the development of Urdu Grammar Checker was done by [1]. In Bangla, it was done by [2] by developing a Bangla Grammar Checker, Punjabi Grammar Checker was propounded by [3] and in Hindi, contribution was extended for checking grammar by [4].

Rule-based [5], statistical (data-driven) [6], and hybrid-based [7] grammar checking methodologies exist. Rule-based categorization is used frequently viz-a-viz, other techniques are used in grammar checking. In this technique, corpus is considered for framing rules as in case of if–then-else rules and given sentence is inputted for checking the accuracy of designed grammar checker. Highlighting aspect of this technique is that such rules are crafted easily and can be modified as and when required. Another motivation for using this feature is that programming is not requisite and a linguistic person can aid the process of rule creation. Additionally, details of the error, if any, are provided easily. Last but not the least, such rules are capable enough to handle basic candid features of specific languages without any major modifications required to entertain input sentence. History of such rule-based systems revolve around languages like Dutch [8], Slavic [9], English [10–13], Punjabi [14], Swedish [15–20], German [21], Korean [22], Danish [23], French [24, 25], Portuguese [26], Persian [27], Afan Oromo [28], Chinese [29], Malay [30].

In statistical grammar checker, annotated corpus is being used and implemented which is obtained from different journals, magazines, or documents. Rules for this system are manually generated. Correctness of a sentence is validated through a thumb rule. A given sentence is passed through a rule to check its correctness.

On success, it is processed against a grammar checker with the help of corpus. On successful pass, the sentence is termed as grammatically correct otherwise it is flagged as a grammatical error. In case of supervised learning, from the given sample, rules are framed as production rules and are used to check the accuracy of the given sentence. The latter technique is infested with a drawback as it is very difficult to perform the task of detecting and recognizing an error in sentence or system.

An alternative approach consists of an Hybrid implementation which comprises Rule-Based and Statistical Grammar Checking which result in a more robust environment and having higher efficiency.

This paper has been organized into the following segments: Segment 2 presents literary aspects of computational linguistics and existing rule-based Punjabi Grammar Checker. Segment 3 presents the critical analysis and shortcomings of existing techniques in light of various sentences procured from standardized organizations and corpus like CDAC, TDIL, Language Newspapers, Texts, etc. Segment 4 presents a novice model to critically justify an advanced Punjabi Grammar checker. Finally, Segment 5 brings our paper to a close and suggests some areas for future investigation.

2 Existing Punjabi Grammar

An interesting aspect of prevailing Grammar Checker is that it follows purely Rule-based philosophy and has no correlation with Statistical approach for computation task, i.e., exhausted hand-crafted rules are followed. These rules can be easily edited and we can add new rules also, further already existing rules can be deleted as and when required based on the concept of production rules written by a linguistic expert without any specific intervention by the programmer.

In the current system, for evaluating correctness of a sentence, Input is given to the Grammar checker, which in turn identifies the end of a sentence with the help of punctuation and breaks down input into unit form, i.e., tokenization and detection of phrases is done here [31].

In preliminary phase, data pre-processing is done. Pre-processing checks for the presence of phrases and tokenizes the sentence into individual words. Once, this process is completed, the checker performs activities like Morphological Analysis (MA), Part-of-Speech (POS) tagging, Error Detection, and Correction. This rule-based approach analyzes the language at Morphological and Syntactical levels. The Morphological Analyzer analyzes each input word and grammatical information is assigned as part-of-speech tags. The suggestions generated for detecting grammatical errors use root word of a particular word along with a full form lexicon. The Part-of-Speech Tagger and Phrase Chunker again follows Rule-Based approach. Phrase Chunker helps in grouping based on predefined phrase chunking rules. Henceforth, at sentence level, rules are applied to check grammatical errors. Excerpt from the system is narrated as follows:

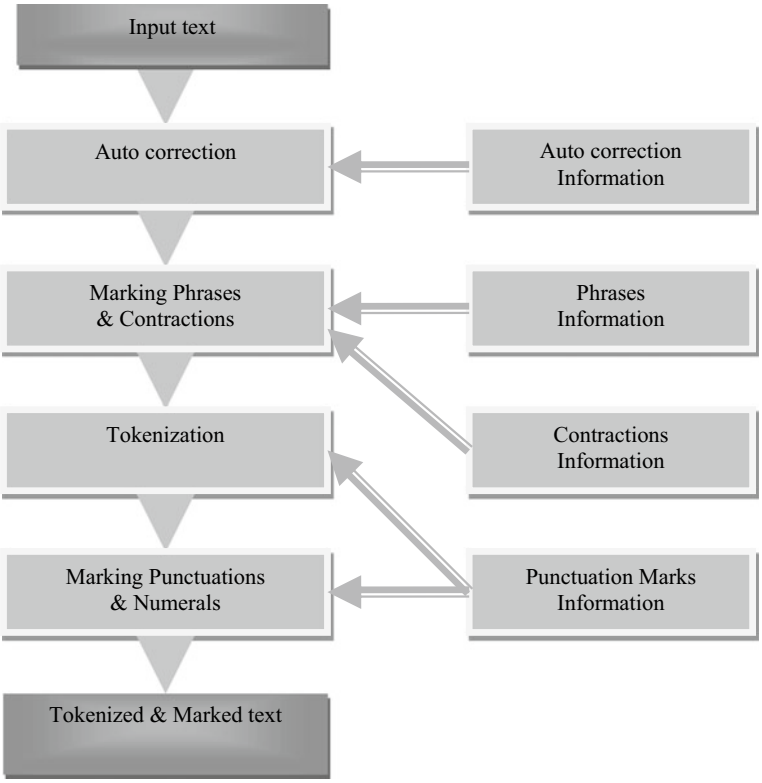

Fig. 2 Pre-processing system design

2.1 Pre-Processing Phase

In the preliminary phase, a Punjabi text is given as input which helps in tokenization, identification of punctuation symbols, detection of contractions, identification of colloquial and phrases, if any. Basically, this phase prepares the input text for next phase, i.e., for morphological analysis as shown in Fig. 2.

2.2 Morphological Analyzer

With the help of full form lexicon concept, possible tags of all words (from the given extract) are assigned. Certain classes like noun, adjective, pronoun, verb, adverb, conjunction, interjection, postposition, ordinals, cardinals, etc., (twenty two in total) are used for classification as per Punjabi grammar. Adjectives are categorized into inflected and uninflected. Similarly, pronoun is classified as personal, interrogative,

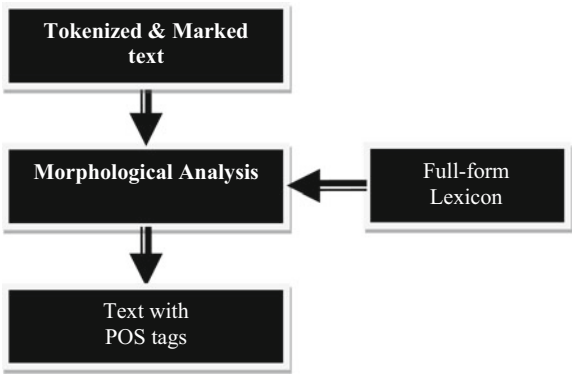

Fig. 3 Morphological analyzer flow diagram

demonstrative, relative, reflexive, and indefinite; verb is classified as main verb, auxiliary verb, and operator verb, respectively. Additionally, details like number, gender, tense, etc., are added depending on the word class. It's worthwhile to mention here that lexicon used for this analyzer is based on full-form, i.e., all common words from literature are stored with their respective root and relevant grammatical information as shown in Fig. 3.

2.3 POS Tagger

In case of disambiguation, i.e., assigning multiple tag to a single word, a Rule-Based POS tagger (parts of speech) has been used to remove this anomaly. Current system uses 600 plus tag sets. Word-specific tags are additionally used. In addition to this, some tags are also there. For instance a notation, NMSD means a noun that is masculine, singular, and direct. In the absence of any statistical corpus used, existing system uses only rule-based phenomenon. The rules are followed in sequential order as shown in Fig. 4.

2.4 Phrase Chunker

Based upon certain phrase chunking rules, grouping of texts is done into various phrases. A rule-based protocol is followed here. Different tag sets are used for different cases—like direct or indirect. Polarity of a sentence, i.e., meaning of a sentence is also considered for framing such rules as shown in Fig. 5.

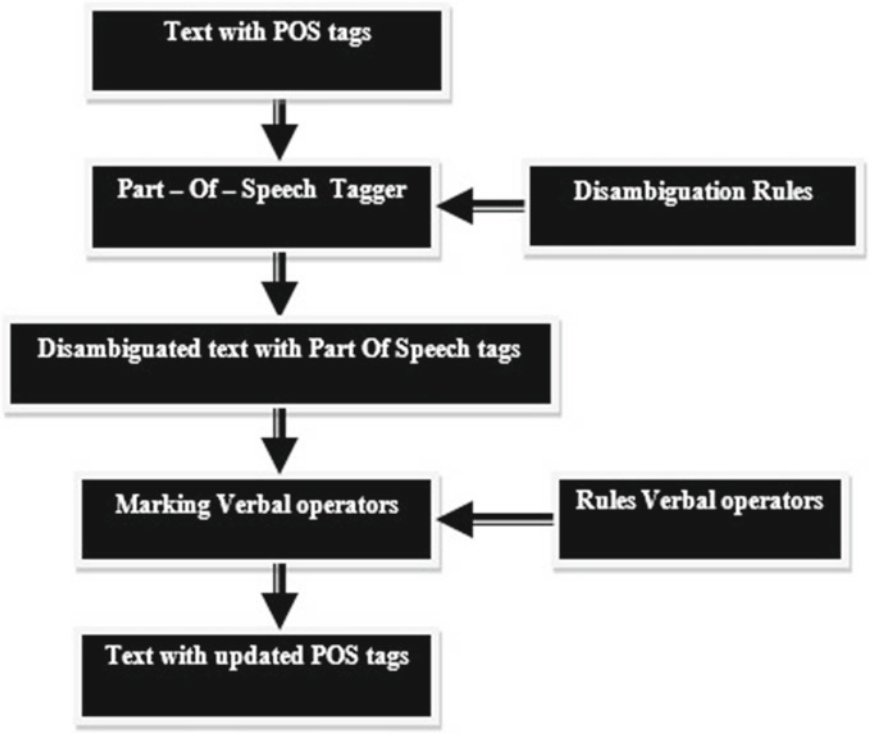

Fig. 4 POS Tagger flow diagram

3 Error Checker and Corrections

In this phase, rules keeping into consideration grammatical errors in phrases and sentence level agreement are implemented. Relevant corrections are suggested on the basis of contextual information on occurrence of error, if any. Subsequently in Grammar Checking phase, error detection rules (rule based) are used to detect potential errors and corrections are provided to resolve such errors.

The concept is summarized as shown in Fig. 6.

4 Critical Analysis and Shortcomings

Existing Punjabi Grammar checker detects grammatical mistakes only for simple sentences and lacks support for compound and complex sentences and raises false alarms. It does not have any component for unknown word guessing. Further, it has a limited domain for certain words that affect its precision and recall. Moreover, Spell checking is not available. Also, the structure lacks support for other languages of

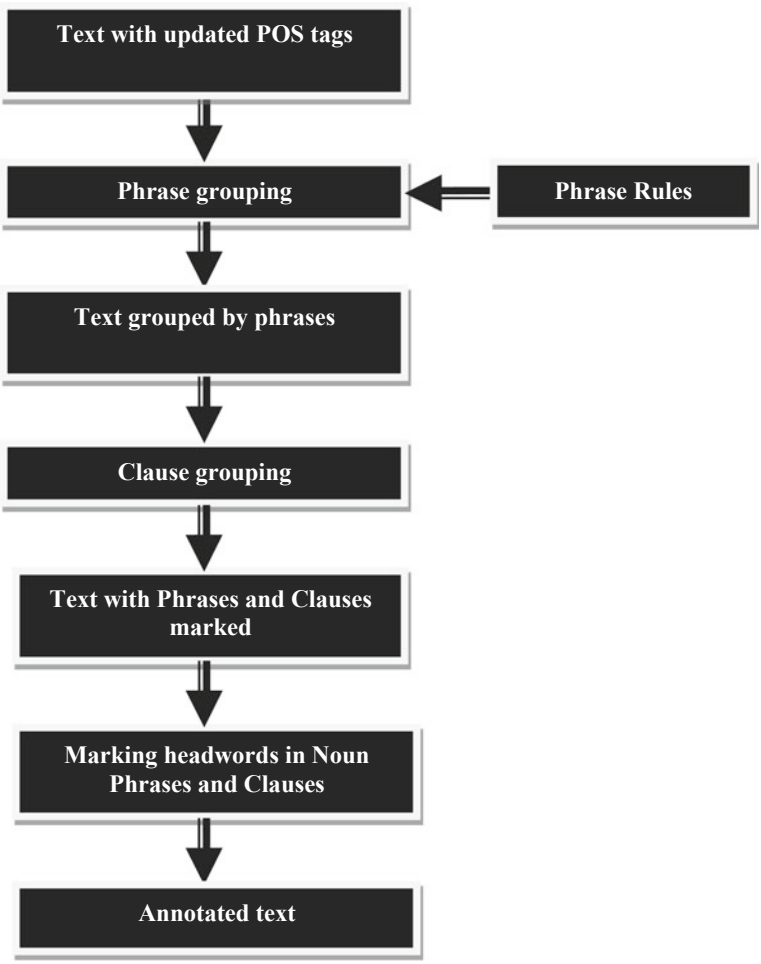

Fig. 5 Phrase chunker flow diagram

Modern Indo-Aryan family, like Hindi, Bengali, etc. [32]. The distinct features of such languages are highlighted in the following Table 1.

Similar theories were put forwarded for other languages including European ones [35–39]. Existing Punjabi Grammar Checker system is processed against sufficient number of sentences (seventy five in total) collected from a standardized repository (as stated earlier) and the results were disappointing. Chosen sentences are processed at the listed URLs:

a. http://punjabi.aglsoft.com/
b. http://pgc.learnpunjabi.org/

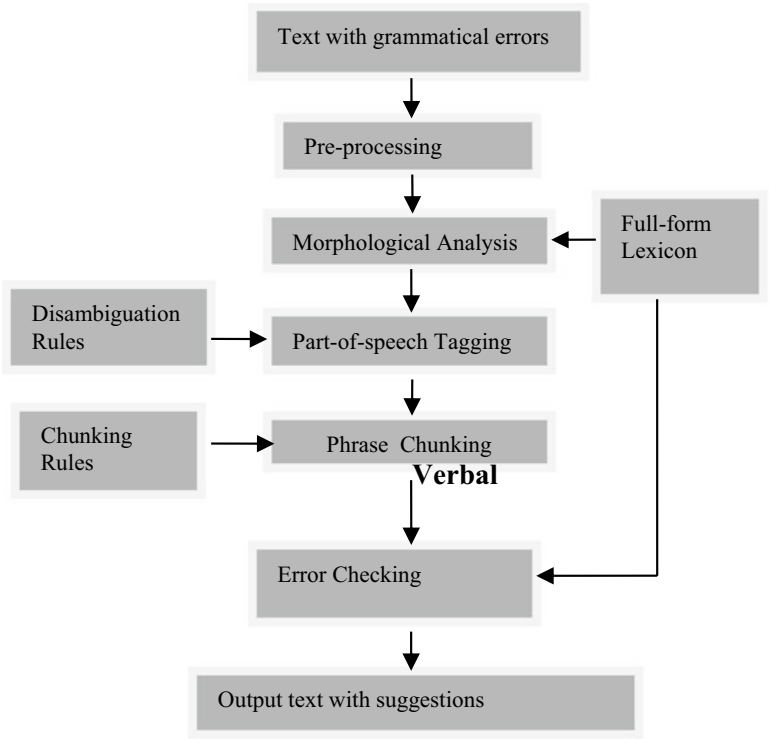

Fig. 6 Model of existing Punjabi grammar checker

Table 1 Analysis of Indo-Aryan languages

Language	Methodology adopted for checking grammar	Characteristics	Evaluation features	Shortcomings
Hindi [4]	Rule-Based	Rich in inflection	Optimal result	Not suitable for Compound and Complex Sentence
Nepali [33]	Rule-Based	Language primitives are shared by Bangla and Hindi	Providing information about errors in simple sentences	Not suitable for Complex and Compound sentences
Urdu [34]	Rule-Based	Formulates S–O-V agreement	Provides error correction by checking structure and grammar	Lacks disambiguation due to Morphology and POS
Bangla [2]	Data-Driven-Based	Formulates agreement of Word	Provides better result	Not suitable for Compound sentences

The analysis report comprises the count of total number of errors (including false alarm) creeping from individual phases of the Grammar Checker and helps us in visualizing the inefficiency of individual components of the Grammar Checker [40–43]. The report is projected through the listed Table 2.

The component-wise reasons for such errors /issues may be accounted for listed factors:

a. In context of a Punjabi sentence, modifiers must collaborate with the noun and modify with respect to gender, number, and case.
b. In Noun-Adjective agreement, Noun needs to be changed sometimes and not only adjective. In current rule, adjective is always changed.
c. POS was not able to remove ambiguity and acted in contrary to its defined assignment and followed the same result of MA.
d. Whenever a word is encountered whose root is not traced, "unknown" tag is assigned.

Table 2 Analysis of Punjabi sentences

Existing grammar checker component	Total number of contributing errors	Percentage contribution	Interpretation (Grammatically correct/incorrect/false alarm)
Morphological Analyzer (MA)	44	58.13	Problem in morph due to an unknown word
Part-of-speech Tagger (POS)	20	26.74	Ambiguity of words is not removed
Phrase Chunker (PC)	5	6.67	In Noun-Adjective agreement, Noun needs to be changed sometimes and not only adjective. In current rule, adjective is always changed
Error Checker (EC)	12	16	Problem in Grammar Component although all previous components were fine and thus Logically Incorrect interpretation
MA and POS	21	28	POS was not able to remove ambiguity and same result of MA is followed
POS and EC	5	6.67	Logically Incorrect interpretation. POS was not able to remove ambiguity

5 Proposed Framework for Punjabi Grammar Checker

All listed shortcomings as stated above may be overcome by using hybrid technique by combining grammar rules with machine learning technique [44]. Till now hybrid approach has not been used for development of Punjabi grammatical error detection because of unavailability of standard Punjabi corpus to be used for machine learning [45]. Two step approach may be followed for the same.

a. Step One

The working of each component of Existing Rule-based System is studied through the listed flow of steps.

As shown in Fig. 7, once an incorrect Punjabi sentence will be given as input, efficiency would be calculated phase-wise, i.e., efficiency would be calculated after MA, Tagging, Chunking, Error Detecting, and Error Correcting, respectively, for analysis so as to evaluate accuracy of each component.

b. Step Two

The components that are responsible for false alarm are identified, and a proposed algorithm to improve these components is followed using two phases. For evaluating a component accountable for false alarm situation, 2-phase process would be followed.

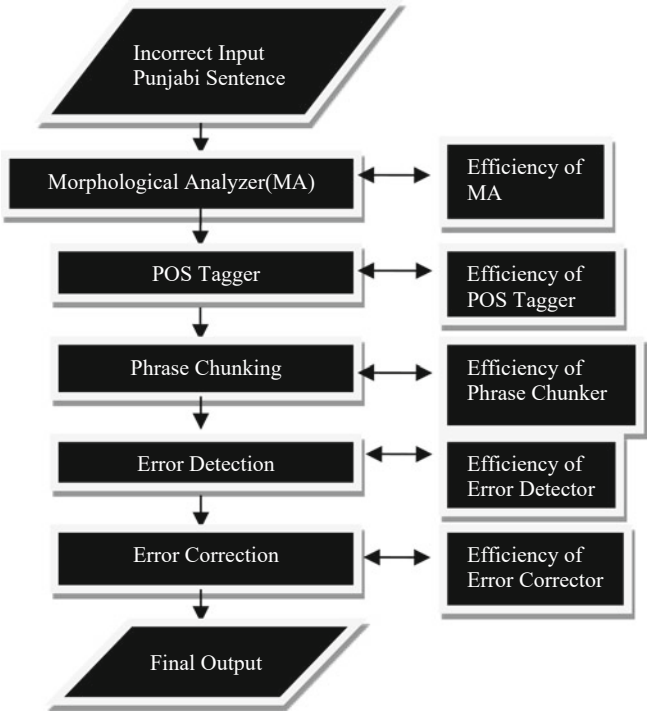

Fig. 7 Proposed model for measuring accuracy

In phase 1, Grammar Checker will perform preliminary check with the help of certain rules. An incorrect sentence would be made to pass through phase II. In phase II, output from phase I would pass through each component (step) to check whether the said component is faulty or not. A particular component is faulty, if the output from that component is incorrect; otherwise, the output will be made to pass through the next step and so on. The step-by-step approach is described in Fig. 8.

6 Results and Discussions

Onto a repository of corpus collected from various standard texts, authorized resource centers like TDIL, etc., as discussed above, we were able to identify Morphological Analyzer as the component contributing maximum in generation of errors, false alarms followed by POS Tagger. The percentage contribution of these were 58.13% and 26.74%, respectively, on individual basis and combined error percentage is 28. Hence, paving a way for further research in this area as these being the important and preliminary steps in overall procedure would be helpful for checking grammatical errors with much accuracy once rectified.

7 Conclusion and Future Work

Our paper has categorically analyzed the accuracy of each component of existing rule-based Punjabi Grammar Checker. The effect of each component is analyzed as it has an implication on the overall accuracy of the system. The parameters for measuring the same were taken as Recall and Precision. This paper also proposes a "Fault Determination System" with an aim of evaluating the "Faulty Component" by following a two-phase approach and concludes with providing the facts and results that Morphological Analyzer and POS Tagger were the faulty components generating false alarms and errors to the tune of 58.13% and 26.74% respectively.

Based on these detections, further research can be carried out for developing a model to overcome these ambiguities using Machine Learning techniques by inculcating a "Hybrid" mechanism. Such "Hybrid" framework may be used for other morphologically rich Indian languages like Oriya, Sanskrit, Hindi, Bengali, etc., and can be further extended for various Natural Language Processing (NLP) tasks associated with Punjabi and other languages.

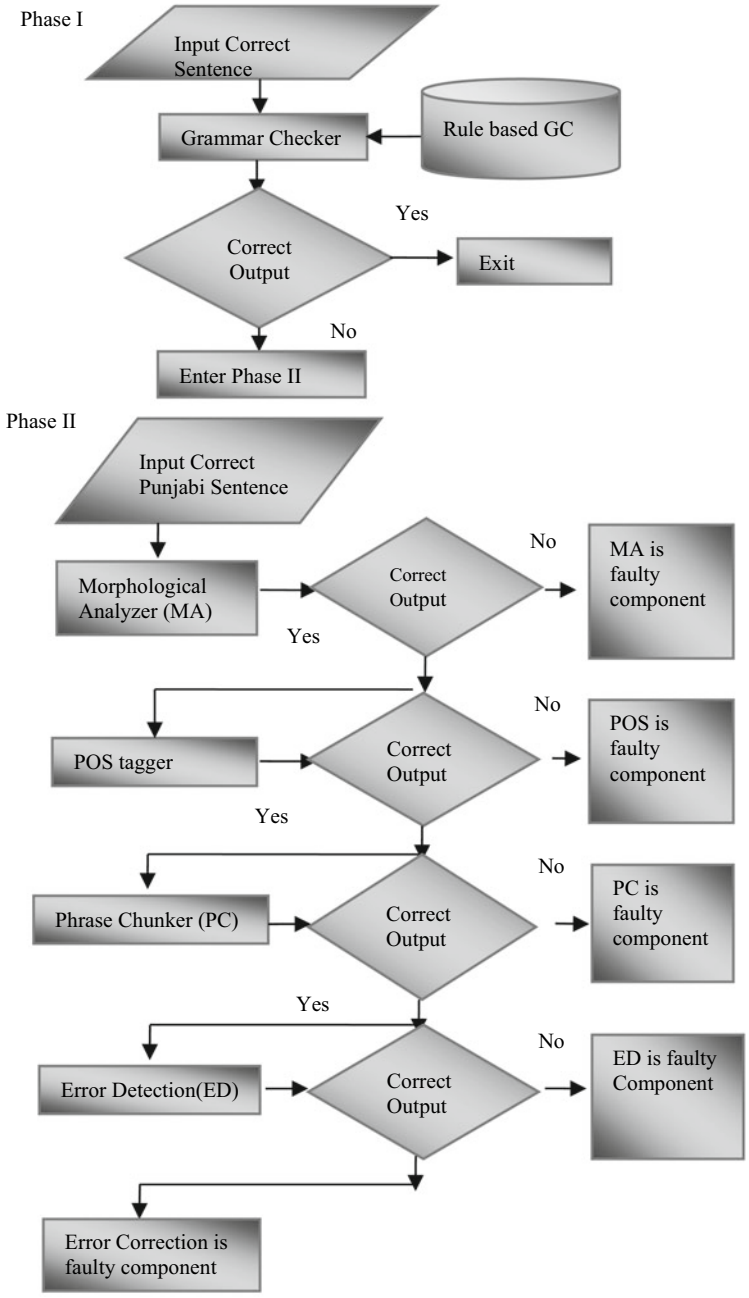

Fig. 8 Proposed model for evaluating faulty component

Acknowledgements We're thankful to DAV University Jalandhar for giving a platform by providing research lab and sharing resources for carrying up our research.

References

1. Bhirud NS (2017) GC for natural languages: a review. IJNLC 6
2. Alam M, UzZaman N, Khan M (2007) N gram based statistical GC for Bangla and English
3. Gill MS, Lehal GS (2008) A GC system for Punjabi. In Coling 2008: companion volume: Demonstrations. pp 149–152
4. Bopche L, Kshirsagar, Dhopavkar G (2011) December. GC system using rule - based morphological process for an Indian language. In ICCCS (International Conference on Computing and Communication Systems), Springer, Berlin, Heidelberg, pp 524–531
5. Naber D.: A rule - based style and GC. Master's thesis, Bielefeld University, (2003)
6. Manchanda B, Sharma SK, Athavale VA (2016) Various technique used for GC. IJCAIT 9(1):177–181
7. Sharma SK, Lehal GS (2016) Improving existing Punjabi GC. In 2016 ICCTICT, pp 445–449, IEEE
8. Vosse T (1992) Detection and correction of Morpho—Syntactic error in real texts. In Third conference on ANLP, pp 111–118
9. Platek M, Kubon V (1994) A Grammar checking of free word order languages based on Grammar based approach. In COLING 1994 Volume 2: The 15th ICCL (International Conference on Computational Linguistics), vol.2
10. Adriaens G (1994) The LRE SECC project: simplified english grammar and style correction in an MT framework. In Proceedings of LEC (language engineering convention) (Paris la Défense [Puteaux], pp 1–8
11. Lin NY, Mar Soe K, Thein NL (2011) Developing a chunk-based grammar checker for translated English sentences. In Proceedings of the 25th Pacific Asia conference on language, information and computation, pp 245–254
12. Paggio P (2000) Grammar and spelling correction for Danish in SCARRIE. In Sixth ANLP (Applied Natural Language Processing Conference), pp 255–261
13. Rider Z (2005) POS tagging and rules matching for Grammar checking. In Class of 2005 SCNLP (Senior Conference on Natural Language Processing)
14. Singh G (2008) Development of Punjabi Grammar Checker. Phd. Dissertation
15. Hein AS (1998) Grammar checking using a chart-based framework. Initial studies. Proceedings of the 11th Nordic conference of CL (NODALIDA 1998), pp 68–80
16. Vandeventer A (2001) Grammar checker creation for CALL by constraint relaxation: a feasibility study. ReCALL 13(1):110–120
17. Carlberger J, Kann V, Domeij R, Knutsson O (2001) The performance and development of Swedish Grammar Checker: A language engineering perspective. NLE 1(1)
18. Fliedner G (2002) NP agreement system for German texts checking. In Proceedings of the ACLSRW, pp 12–17
19. Bigert J, Kann V, Knutsson O, Sjobergh J (2004) Grammar checking for Swedish second language learners
20. Hashemi SS (2003) Automatic detecting of grammar errors texts of children's in primary school. A finite state approach (Doctoral dissertation, University of Goteborg)
21. Schmidt - Wigger A (1998) Style checking and Grammar for German. In Proceedings of CLAW, Vol 98, pp 76-86
22. Chae YS (1998) Korean proofreading system Improvement using collocation and corpus. In Proceedings of the Korean society for language and information conference, KSLI, pp 328–333

23. Paggio P (2000) April. Grammar correction and Spelling checking for Danish in SCARRIE. In Sixth ANLPC (Applied Natural Language Processing Conference), pp 255–261
24. Music B, Helfrich A (2000) Design and evaluation of grammar checkers in multiple languages.In: The 18th International Conference on Computational Linguistics (ICCL), In COLING 2000, vol 2
25. Vandeventer A (2000) Creation of a grammar checker for CALL using constraint relaxation: a feasibility study. ReCALL 13(1):110–120
26. Kinoshita J, do Nascimento Salvador L, de Menezes (2006) C.E.D.: Brazilian-Portuguese Grammar Checker based on the CETENFOLHA Corpus (CoGrOO). In LREC, pp 2190–2193
27. Ehsan N, Faili H (2010) Towards Persian language Grammar Checker development. In Proceedings of the 6th international conference on natural language processing and knowledge engineering (NLPKE-2010), pp 1–8, IEEE
28. Tesfaye D (2011) A rule-based Grammar Checker for Afan Oromo. IJACSA Editorial 2(8):126–130
29. Jiang Y, Wang T, Lin T, Zhang W, Wang F, Cheng W, Liu X, Wang C (2012) A Chinese spelling and grammar detection rule - based system utility. In 2012 ICSSE, IEEE, pp. 437–440
30. Kasbon R, Mahamad S, Amran N, Mazlan E (2011) A sentence checker for Malay language. WASJ (Special Issue on Computer Applications and Knowledge Management) 12:19–25
31. Gill MS, Lehal GS, Joshi SS (2008) A Punjabi grammar checker. In: Proceedings of the 3rd international joint conference on NLP, vol 2
32. Kitchenham B (2010) Whats up of software metrics?–A preliminary mapping study. JSS 83(1):37–51
33. Bal BK, Shrestha P, Pustakalaya MP, PatanDhoka N (2007) Architectural and system design of the Nepali GC (grammar checker). PAN Localization Working Paper
34. Kabir H, Nayyer S, Hussain S, Zaman J (2002) Urdu GC using two pass parsing implementation. In: Proceedings of IEEE international multi topic conference, pp 1–8
35. Callison-Burch C, Koehn P, Monz C, Zaidan O (2011) Findings of the 2011 workshop on SMT. In: Proceedings of the sixth workshop on statistical machine translation, pp 22–64
36. Mittal M, Kumar D, Sharma SK (2016) Grammar checker for Asian languages: A survey. IJCAIT 9(1):163
37. Sharma SK (2017) Error detection in english and other european languages using statistical approaches. AGU IJRSSH 5:234–239
38. Sharma SK (2016) Rule - Based Grammar Checker System (A Survey). IJCAIT 10(1):217–220
39. Schmaltz A, Kim Y, Shieber SM, Rush AM (2017) Sentence correction using adapting sequence models. arXiv preprint arXiv: 1707.09067
40. Rashmi S, Hanumanthappa M (2017) Qualitative and quantitative study of syntactic structure: a GC using part of speech tags. IJIT (International Journal of Information Technology) 9(2):159–166
41. Sharma SK, Lehal GS (2011) HMM to improve the accuracy of Punjabi POS tagger. In: 2011 IEEE ICCSAE (International Conference on Computer Science and Automation Engineering), IEEE, vol 2, pp 697–701
42. Bharti SK, Babu KS (2017) Automatic keyword extraction for text summarization: a survey. arXiv preprint arXiv: 1704.03242
43. Deksne D, Veisbergs A (2018) A workflow for supplementing a Latvian-English Dictionary with data from parallel corpora and a reversed English-Latvian Dictionary. In: Proceedings of the XVIII EURALEX International Congress: Lexicography in Global Contexts, pp 127–135
44. Wang Y, Shih C (2018) A hybrid approach combining statistical knowledge with CRF (Conditional Random Fields) for Chinese grammatical error detection. In: Proceedings of the 5th workshop on NLP (Natural Language Processing) techniques for educational applications, pp 194–198
45. Jindal L, Sharma SK, Singh H (2021) Framework for grammatical error detecting and correcting system for punjabi language using stochastic approach