A Comparative Analysis of Automated Grammar Checking Techniques

Madhvi Soni[1], Jitendra Singh Thakur[2]

Dept. of Computer Science & Engineering, Jabalpur Engineering College, Jabalpur, Madhya Pradesh 482011, India.

[1]madhvi.soni21@gmail.com , [2]jsthakur@jecjabalpur.ac.in

Article Info

Page Number: 19 - 26

Publication Issue:

Vol 70 No. 1 (2021)

Article History

Article Received*: 12 January 2021*

Revised*: 25 February 2021*

Accepted*: 20 April 2021*

Publication*: 09 June 2021*

Abstract

Grammar checking has been identified as an important application of Natural Language Processing. Development of automated grammar checking tools is gaining popularity among the researchers as well as among the commercial software developers. There are different techniques which are used for the development of the grammar checker of any language. These techniques include rule based technique, machine learning based technique and hybrid technique. This research article, presents the analysis of these techniques, their working methodology highlighting the benefits and the associated challenges. Our study also investigates over 30 research articles of grammar checking approaches in different languages. This will help our research community to analyze the evolution of the grammar checking task over the past few decades and take the further research and development decisions.

Keywords—Grammar checking, Natural Language Processing, Rule based, Machine Learning based, Hybrid Technique.

INTRODUCTION

Grammar checking software are intelligent computer applications which check the correctness of an input sentence. Correctness of a sentence is checked with the help of an underlying grammar of the natural language. The grammar consists of a set of rules which govern the formation of sentences in that language. It is difficult for the second or foreign learners of a language to write the correct grammatical sentences. Here comes the need of automatic grammar checking tools which could help in language learning or proofreading. A grammar checking tool generally takes some input text, detects the ungrammatical phrase and possibly corrects it automatically. Some of the key features of a grammar checker as described by Naber [3] are (1) it should be faster in response, (2) can be integrated with the existing word processor, (3) should have lower false alarm rate, and (4) it should find almost all types of errors.

The development of automatic grammar checking software was started in the early 80's. The Earliest tools were based on error correction by simple word matching or word replacement [1]. Nowadays, more sophisticated tools have been developed such as Grammarly [2], Ginger[4], BonPatron [30] and LanguageTool [3], which combines a variety of techniques to automatically detect and correct ungrammatical phrases in the text. To develop new tools with enhanced capability, one must seek for the approaches, methodologies and techniques applied in the past.

This paper is organized into following sections: Section II presents the review of the existing literature. Section III presents the classification of Grammar checking techniques. Section IV presents the comparative analysis of these techniques. Finally, section V concludes our study.

II. LITERATURE REVIEW

Bustamante et al[17] developed a GramCheck for Spanish language using the rule based technique. The rules were handcrafted after carefully analyzing the Spanish text from newspapers. Domeij et al [21] developed Granska, a Swedish grammar checker using the rule based technique whereas Jonas et al [26] proposed a new Swedish grammar checker using machine learning technique. Bal Krishna et al [20], Bopche et al [14], Gill et al [18] and Sagar et al[23] developed rule based grammar checker for Nepali, Hindi, Punjabi and Kannada languages respectively. Van Compernolle et al [30], Ehsan et al [16], Tesfaye et al [19], Jiang et al [15] and Matthew Phillip et al [22] developed rule based grammar checkers for French, Persian, Afan Oromo, Chinese and Filipino languages respectively. For developing English Grammar checkers [25], [27], [28], [5], [9], [11] and [13] applied machine learning technique, while [31],[32], [3], [33], [8], [10] applied the rule based. A Bangla Grammar checker was developed by Alam et al [24] using the machine learning technique. Hybrid Systems were developed for English language by [6], [29] and [12]. No Literature was found for [2] and [4]. A comparative summary of these Grammar checkers is given in Table 1, Table 2 and Table 3.

III. CLASSIFICATION OF TECHNIQUES

After carefully analyzing the literature, we identified that all the methods can broadly be classified into three main techniques; namely Rule based grammar checking, Machine learning based Grammar checking and Hybrid technique of grammar checking. See Figure 1.

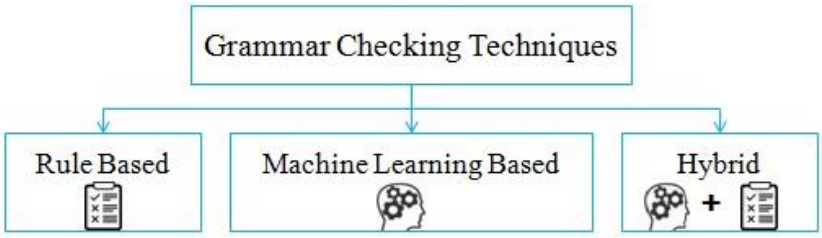

Figure 1: Classification of Grammar Checking Techniques

1) Rule based technique:

The traditional approach of grammar checking is to detect the ungrammatical sentence by checking the text against a finite set of handcrafted grammar rules. If the text does not match any of the rules from the rule set then the text is marked as erroneous. These rules can be utilized to provide comments or explanation of why the sentence is ungrammatical. This technique is extremely helpful for the purpose of Computer Assisted Language Learning [34] (CALL). Rules can be easily added to correct new errors or deleted/modified to remove false alarms. However, manual maintenance of hundreds of grammar rules is a complex task.
A list of rule based grammar checkers developed for different languages is provided in table 1.

2) Machine Learning based technique:

Machine learning technique is based on the statistical analysis of the text from large corpora for automatically detecting and correcting grammar errors [5], [13]. These corpora are built by collecting correct text from the native or non-native speakers of a particular language. Text could also be collected from newspapers, documents, books, essays and other such resources. The text is then tagged using Part-of-speech tag list. These tags or some other features of correct sentences from the corpus are utilized to train the machine learning model. The model automatically corrects an ungrammatical phrase based on the learned patterns. This technique is not Language dependent since it does not require knowledge of deep grammar rules of any language. This technique is easier to implement as compared with rule based but the availability of large corpus (annotated as correct and incorrect sentences) makes it unsuitable for low resource languages.

A list of machine learning based grammar checkers developed for different languages is provided in Table 2.

3) Hybrid technique:

Hybrid techniques [6], [29], [12] are an amalgamation of machine learning and rule based techniques of grammar checking. The text from annotated corpus is used to train the learning model and the results could be refined by applying manually designed rules which are language specific [6]. This technique utilizes the best advantages of each technique and hence is able to detect and correct more complex type of errors. Introducing the statistical analysis in rule based system reduces the complex and mundane task of rule designing. As it combines both the techniques, the developed system is more robust and achieves higher efficiency.

A list of hybrid grammar checkers developed for different languages is provided in Table 3.
.

Table 1: Rule Based Techniques.

Approach	Year	Language	Dataset used in the research	Results
[17]	1996	Spanish	70,000 words including text fragments from literature, newspapers	Not specified
[31]	1997	English	Students essay	Not specified
[32]	1997	English	Corpus of 27000 words of text by French native speakers.	Not specified
[21]	2000	Swedish	Swedish sentences	Not specified
[3]	2003	English	Mailing list error corpus of 224 sentences.	Not specified
[33]	2004	English	SST corpus of 221 sentences	Success rate: 80%
[8]	2007	English	Reuters-21578 corpus, sentences from book-Avoid Errors	Not specified
[20]	2007	Nepali	Nepali Text	Not specified
[30]	2007	French	Corpus of French text.	Accuracy: 86%
[18]	2008	Punjabi	Punjabi sentences in Gurmukhi script	Not specified
[23]	2009	Kannada	Kannada sentences	Not specified
[16]	2010	Persian	Persian sentences	Precision: 71% Recall: 83%
[10]	2010	English	Corpus of English sentences collected from lang-8.com.	Success rate: 67.2% Precision: 40.16% Recall: 20.28%

[14]	2011	Hindi	Hindi sentences.	Not specified
[7]	2011	English	Not specified	Not specified
[19]	2011	Afan Oromo	Graduate student's thesis text in Afan Oromo	Precision: 88.89% Recall: 80.00%
[15]	2012	Chinese	Chinese Wikipedia Errors from China Matriculation Examinations and Chinese Books	Accuracy: 90%
[22]	2017	Filipino	Filipino sentence corpus	Accuracy: 64%

Table 2: Machine Learning Based Techniques.

Approach	Year	Language	Dataset used in the research	Results
[25]	2005	English	Wall Street Journal (WSJ) corpus, Penn Treebank 3 release, BNC sampler, SUSANNE, MULTEXT-East	Not specified
[26]	2006	Swedish	Swedish Parole corpus, Swedish text from Internet	Precision: 92%
[27]	2006	English	Japanese Learners' English corpus	Accuracy: 88.7%
[28]	2006	English	WordNet	Not specified
[5]	2006	English	Reuters newswire articles, CLEC corpus. English sentences from Chinese websites.	Success rate: 61.81 %
[24]	2007	Bangla, English.	Bangla sentences from 5000 words Prothom-Alo corpus. English Sentences from Brown Corpus.	Accuracy: 63% (English) 53.7% (Bangla)
[9]	2007	English	Hiroshima English Learners corpus, Japanese Learners of English corpus & Chinese Learner Error corpus.	Accuracy: 81.3 % Precision: 83.09 % Recall: 81.24 % F-score: 81.25%
[11]	2011	English	NUCLE corpus, Gigaword Corpus, Wall Street Journal.	F score: 19.29 (articles) 11.15 (Prepositions)
[13]	2013	English	NUCLE, Google web 1T 5-gram corpus.	Precision: 62.19% Recall: 31.87% F score = 42.14%

Table 3: Hybrid Techniques.

Approach	Year	Language	Dataset used in the research	Results
[6]	2007	English	MetaMetrics corpus of 1100 & 1200 Lexile text, newspaper text, Chinese, Japanese & Russian's ESL essays.	Precision: 80% Recall: 30.4 %
[29]	2010	English	English sentence corpus	Not specified

[12]	2013	English	NUCLE, CLC, FCE, EVP corpora	Precision: 46.70 % Recall: 34.30 % F score: 43.55 %

IV. COMPARISON OF GRAMMAR CHECKING TECHNIQUES

In this section, we present the comparison of the three grammar checking techniques. Each technique has its own benefits and drawbacks.The selection of the technique depends upon the underlying language and its features and also to the available language resources to some extent. Table 4 provides comparison of the three techniques.

Table 4: Comparison of Grammar Checking Techniques.

Technique	Benefits	Challenges
Rule Based	Rule based systems are easy to build.	Rule designing needs a lot of manual effort.
	Can provide proper explanation or the erroneous sentence.	Requires complete number of grammar rules to cover all types of errors.
	Rules can be easily added, modified or removed.	Requires deep linguistic knowledge.
	It is easy to incorporate domain knowledge into linguistic knowledge.	Complexity of the rule increases exponentially as we try to solve different types of errors.
	The rules can be written by the linguists, having limited or no programming skills.	Difficult to implement automatic error correction.
	The rules designed for one system can be reused for another similar system.	
	It is easy to trace the rule which made the decision.	
Machine Learning Based	Provides better results as compared to rule based systems.	ML based systems uses corpus data which must contain sufficient instances of all types of grammar errors.
	Helpful in addressing a wide range of complex errors	Explanation of errors or comments cannot be provided.
	Deep knowledge of the underlying language grammar is not necessary.	System may predict a correct sentence as wrong. (False Alarm)
	Training data is easily available.	Results of ML based systems are difficult to interpret.
	Language independent system can be developed.	ML based systems requires annotated corpus, which is a problem with low resource languages.
	A variety of learning models can be applied.	Corpus must be large enough for better learning.
	Automatic error correction can be implemented easily.	
Hybrid	Helpful in addressing a wide range of complex errors.	Complex systems.
	Robust and efficient.	Needs careful experimental analysis to identify which part of the system must be rule based and which must be ML based.

	Introduction of ML reduces the number of rules to be designed.	
	Some hybrid systems reported better performance.	

V. CONCLUSION

Development of Grammar checking tools has been evolved since 1980. There are three development techniques namely rule based technique, machine learning based technique and hybrid technique. Rule based technique utilizes handcrafted rules to identify grammatical errors in the text, Machine learning technique utilizes a large corpus of text to learn the correct sentences and the hybrid one is the combination of both the techniques. Each of these techniques has their own benefits and drawbacks. Hence, the choice of a technique for developing a grammar checker depends on the target language and its resources.

Much work has been done in developing grammar checkers for different languages. Some of them are low resource languages which are not widely used; therefore they lack the availability of large corpora for the purpose of research and development in the field of grammar checking. But, the most progressive work has been done for English language, so far. Out of the 30 research articles that we have analyzed, 18 were based on English grammar checking. Most of them have achieved high accuracy. Although, there is much need for improvement. Thus, future studies can be done on analyzing the approaches of automatic grammar checking for English language and identifying the current state-of-art.

REFERENCES

[1] Dale, Robert. "Checking in on grammar checking." *Natural Language Engineering* 22.3 (2016): 491-495.
[2] www.grammarly.com
[3] Naber, Daniel. "A rule-based style and grammar checker." (2003).
[4] www.gingersoftware.com
[5] Brockett, Chris, William B. Dolan, and Michael Gamon. "Correcting ESL errors using phrasal SMT techniques." *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2006.
[6] Chodorow, Martin, Joel R. Tetreault, and Na-Rae Han. "Detection of grammatical errors involving prepositions." *Proceedings of the fourth ACL-SIGSEM workshop on prepositions*. Association for Computational Linguistics, 2007.
[7] Mozgovoy, Maxim. "Dependency-based rules for grammar checking with LanguageTool." *Computer Science and Information Systems (FedCSIS), 2011 Federated Conference on*. IEEE, 2011.
[8] Kumar, Akshat, and Shivashankar B. Nair. "An artificial immune system based approach for English grammar checking." *Artificial immune systems*. Springer Berlin Heidelberg, 2007. 348-357.

[9] Sun, Guihua, et al. "Detecting erroneous sentences using automatically mined sequential patterns." *ACL*. 2007.

[10] Huang, An-Ta, et al. "Discovering Correction Rules for Auto Editing." *Computational Linguistics and Chinese Language Processing* 15.3-4 (2010): 219-236.

[11] Dahlmeier, Daniel, and Hwee Tou Ng. "Grammatical error correction with alternating structure optimization." *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011.

[12] Felice, Mariano, et al. "Grammatical error correction using hybrid systems and type filtering." Association for Computational Linguistics, 2014.

[13] A. Rozovskaya, K.-W. Chang, M. Sammons, D. Roth, The university of illinois system in the conll-2013 shared task, in: Proceedings of the Seventeenth Conference on Computational Natural Language Learning: Shared Task, 2013, pp. 13–19

[14] Bopche, Lata, Gauri Dhopavkar, and Manali Kshirsagar. "Grammar Checking System Using Rule Based Morphological Process for an Indian Language." *International Conference on Computing and Communication Systems*. Springer, Berlin, Heidelberg, 2011.

[15] Jiang, Ying, et al. "A rule based Chinese spelling and grammar detection system utility." *2012 International Conference on System Science and Engineering (ICSSE)*. IEEE, 2012.

[16] Ehsan, Nava, and Heshaam Faili. "Towards grammar checker development for Persian language." *Proceedings of the 6th International Conference on Natural Language Processing and Knowledge Engineering (NLPKE-2010)*. IEEE, 2010.

[17] Bustamante, Flora Ramírez, and Fernando Sánchez León. "GramCheck: A grammar and style checker." *Proceedings of the 16th conference on Computational linguistics-Volume 1*. Association for Computational Linguistics, 1996.

[18] Gill, Mandeep Singh, Gurpreet Singh Lehal, and Shiv Sharma Joshi. "A punjabi grammar checker." *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II*. 2008.

[19] Tesfaye, Debela. "A rule-based Afan Oromo Grammar Checker." *IJACSA Editorial* (2011).

[20] Bal, Bal Krishna, et al. "Architectural and system design of the Nepali grammar checker." *PAN Localization Working Paper* (2007).

[21] Domeij, Rickard, et al. "Granska–an efficient hybrid system for Swedish grammar checking." *Proceedings of the 12th Nordic Conference of Computational Linguistics (NODALIDA 1999)*. 2000.

[22] Go, Matthew Phillip, Nicco Nocon, and Allan Borra. "Gramatika: A grammar checker for the low-resourced Filipino language." *TENCON 2017-2017 IEEE Region 10 Conference*. IEEE, 2017.

[23] Sagar, B. M., G. Shobha, and Ramakanth Kumar. "Solving the noun phrase and verb phrase agreement in Kannada sentences." *International Journal of Computer Theory and Engineering* 1.3 (2009): 288.

[24] Alam, Md, Naushad UzZaman, and Mumit Khan. "N-gram based statistical grammar checker for Bangla and English." (2007).

[25] Dickinson, Markus. *Error detection and correction in annotated corpora*. Diss. The Ohio State University, 2005.

[26] Sjöbergh, Jonas. *The internet as a normative corpus: grammar checking with a search engine*. Department of Theoretical Computer Science, Computer Science and Communication, Kungliga tekniska högskolan (KTH), 2006.
[27] Lee, John, and Stephanie Seneff. "Automatic grammar correction for second-language learners." *Ninth International Conference on Spoken Language Processing*. 2006.
[28] Moré, Joaquim. "A grammar checker based on web searching." *Digithum* 8 (2006): 1-5.
[29] Lin, Nay Yee. "Developing a Hybrid Approach for English Grammar Checker." Fifth Local Conference on Parallel and Soft Computing, 2010.
[30] van Compernolle, Rémi A. "Terry Nadasdi and Stéphan Sinclair (developers), BonPatron: An Online Grammar, Spelling, and Expression Checker,© Nadaclair Technologies, 2001–2009, URL: http://bonpatron. com." *Journal of French Language Studies* 19.3 (2009): 406-409.
[31] Park, Jong C., Martha Stone Palmer, and Clay Washburn. "An English Grammar Checker as a Writing Aid for Students of English as a Second Language." *ANLP*. 1997.
[32] Tschichold, Cornelia, et al. "Developing a new grammar checker for English as a second language." *Proc. From Research to Commercial Applications: Making NLP Work in Practice* (1997): 7-12
[33] Bender, Emily M., et al. "Arboretum: Using a precision grammar for grammar checking in CALL." *Instil/icall symposium 2004*. 2004.