

An Approach for Checking Grammar for Telugu Language Compound Sentences

V.Suresh^{#1}, M.S.Prasad Babu^{*2}

[#]Department of Computer Science and Engineering, JNTU Kakinada University

^{*}Department of Computer Science and Systems Engineering, Andhra University

Vignan's Institute of Information Technology, Duvvada, Visakhapatnam, A.P., India

Andhra University College of Engineering, Andhra University, Visakhapatnam, A.P., India

¹vayasisesuresh@gmail.com, ²drmsprasadbabu@yahoo.co.in

Abstract—A Grammar Checking for Telugu Language Compound Sentences is one of the basic applications of the Natural Language Processing. A sentence composed of single independent clause is called a simple sentence and a sentence having more than one independent clause is a compound sentence. Once the sentence is identified as compound or complex sentence, the next step is to identify its pattern. After identification of patterns, various clauses present in the sentence are extracted and grammar checking is performed on them. For grammar checking of compound sentence, it is necessary to identify the structure of these sentences. The structure of compound sentence can be identified on the basis of number of clauses and types of clauses present in them. This study will be helpful in identifying and separating the compound sentences from Telugu language. Also this study will be helpful in developing other Natural Language Processing (NLP) applications like converting a compound sentence in simple sentences, grammar checking of compound sentences.

Index Terms— NLP, Compound Sentences, Independent clause, Dependent clause. (key words)

1. INTRODUCTION

1.1. COMPOUND SENTENCES OF GRAMMAR CHECKING SYSTEM :

Compound sentences are composed of at least two independent clauses joined by coordinate conjunctions,

comma or semicolon. For grammar checking of compound sentences, each clause is extracted from the sentence and grammar checking is performed on it. Since there may be two to any number of independent clauses present in compound sentences, therefore, divide and conquer model can be used for grammar checking of compound sentences. In accordance with divide and conquer, the compound sentence is simplified by splitting it into individual clauses and then each clause undergoes error detection and correction mechanism. In this way, overall grammar checking process for compound sentences takes place in two steps; first step is extracting the independent clauses from the compound sentence and second step is to perform grammar checking on each extracted clause. For extracting the independent clauses from the compound sentence, the clause boundary of these clauses has to be identified.

1.1.1. Types of grammatical errors in compound sentences:

As the compound sentences are composed of more than one independent clauses, therefore each independent clause undergoes detection of following types of errors:

1.1.1.1. All the Noun phrases joined by conjunctions to form group having same case:-

If in an independent clause, there are two or more than two noun phrases (NP) joined by a conjunction, then all these noun phrases (NP) must have same case.

Consider the following example:

Telugu : అబ్బాయిలు మరియు అమ్మాయిలు విందులో పాల్గొన్నారు.

Roman Transliteration : abbayilu mariyu ammayilu vindulo
palgonnaru

English : Boys and girls participated in the function

1.1.1.2 యొక్క (yokka) కలుపుట (kaluputa) should be in agreement with noun phrase next to it in terms of number and gender:

If there is యొక్క (yokka) postposition in a sentence, then this postposition should be grammatically in agreement with noun phrase next to this postposition in terms of number and gender. Consider the following example:

అబ్బాయి యొక్క పుస్తకం దారి పోయింది

(abbayi yokka pusthakam dari poyindi)

Boy's book was lost.

1.1.1.3. Modifier and Noun agreement:

In a noun phrase, the modifier and noun should be grammatically in agreement in terms of number and gender. Consider the following example:

నల్ల అబ్బాయిలు అటలాడుకున్నారు

(nalla abbayilu aataladu kunnaru)

Dark boys were plying

1.1.1.4. Order of modifier of Noun phrase:

In an independent clause, if a noun has two or more modifiers then the order of modifier is fixed. Generally, numerals precede adjective or verb phrases. Consider the following example:

1.1.1.5. Order of word in verb phrase:

The words in the verb phrase should follow the specific sequence i.e. main verb should be followed by an operator and an operator should be followed by an auxiliary verb.

Consider the following example:

బాలుడు స్కూలుకి వెళుతున్నాడు

(baludu skool ki veluthunnadu)

Boy is going to school

1.1.1.6. Style error:

These are the errors which are generated due to missing a punctuation mark, using an in-appropriate punctuation mark or by using duplicate words in a sentence. In this research work, three types of style errors have been covered; one is error due to using an inappropriate punctuation, second error due to use of a duplicate word in a sentence and third is due to missing a punctuation mark. All these errors have been discussed in the following section:

1.1.1.6.1. Error due to using in-appropriate punctuation:

Thus type of error occurs due to use of an incorrect punctuation mark or missing punctuation.

1.1.1.6.2. Error due to missing conjunction or punctuation:

Compound sentences are composed of independent clauses separated by conjunctions. These conjunctions include some punctuation marks like comma (,) or they may be words belonging to coordinate conjunctions word class.

ఆమె లేవక ముందే దొంగతనం జరిగిపోయింది

(aame levaka munude dongathanam jarigipoyindi)

When she wake up, the robbery had already happened

1.1.1.6.3. Error due to duplicate words:

Sometime, while writing a sentence, a word is typed twice. This results in unstructured sentence and it becomes difficult to understand the meaning of the sentence.

2. Checking Grammar for Telugu Language Compound Sentences

The algorithm used in the grammar checking of compound sentences is as following: **Algorithm:**

For error detection in compound sentences:

Databases used: Error type.

In put: incorrect simple sentence (independent clause)

Output: Corrected simple sentence.

1. Get all the error type that have *On Off* value set to 1, from the error type database sorted by the *Priority* field

2. Repeat steps 3 to 5 for current clause.
3. Repeat steps 4 and 5 for all the error types.
4. Call the respective module to perform the correction on the current clause.
5. Output the corrected sentence.

Consider the following incorrect compound sentence:

Incorrect sentence:

రాముడు మంచి బాలురే కాని అతనికి కళాశాలలో ర్యాంక్ రాదు

(ramudu manchi balure kani athaniki kalaselalo rank radu)

Rama is a good boys but he doesn't have rank.

In above incorrect sentence, there are two clauses in the compound sentence and each clause contains errors. The first clause is రాముడు మంచి బాలురే (ramudu manchi balure) and it contains noun modifier agreement error as the modifier రాముడు (ramudu) (singular) does not grammatically agree with noun బాలురే (balure) (plural) in terms of number. Second clause is కాని అతనికి కళాశాలలో ర్యాంక్ రాదు (kani athaniki kalaselalo rank radu), and it contains subject verb agreement error as the object అతనికి (athaniki) (feminine) does not grammatically agree with verb రాదు (radu) (masculine) in terms of gender. Both these errors are detected and rectified in two steps. In the first step, two clauses are separated from the sentence and in the second step, these clauses are detected for the presence of error. After applying detection and correction on individual clauses, the final updated output given by the researcher's system is:

Corrected sentence:

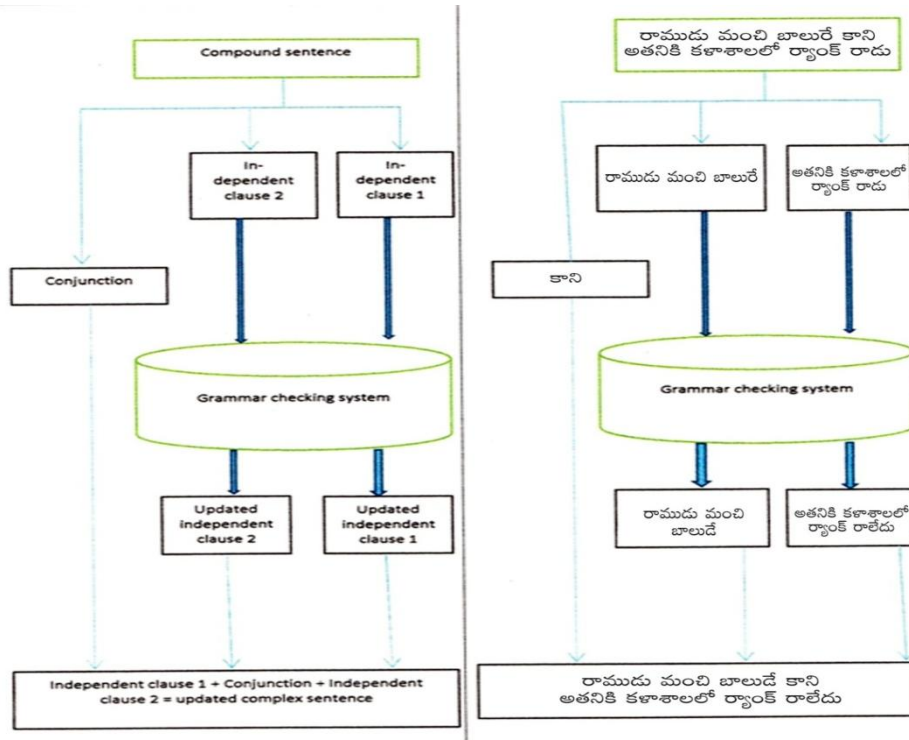
రాముడు మంచి బాలుడే కాని అతనికి కళాశాలలో ర్యాంక్ రాలేదు.

(ramudu manchi balude kani athaniki kalaselalo rank raledu)

Rama is a good boy but he doesn't have rank.

The complete architecture of the above method with example

has been shown below flow chart. It is clear from below flow chart that the compound sentence is first simplified by splitting it at the conjunction and separating each independent clause. Then each independent clause is passed through grammar checking system where error detection and correction algorithm mention above is applied.



3. Results and Discussion

GRAMMATICAL MISTAKES COVERED IN COMPOUND SENTENCES:

In this grammar checking system, following two categories of grammatical errors related to compound sentences have been covered:

3.1. Errors in the structure of compound sentences (style error):

Under this category, errors related to use of inappropriate punctuation marks, repeated/duplicate words, missing comma and missing conjunctions have been covered. Table 3.1 shows the list of these errors with appropriate examples. In this table, error type 1 to 3 lies under inappropriate punctuation marks category.

Table 3.1: Style errors in compound sentences

Sr. No	Type of error	Example	Remarks
1.	Missing sentence Ender	ఇద్దరులు పువ్వులు ఇబట్టి (clasulu unnayi kabatti)	The sentence sender is missing with కళాశాల వచ్చాను (kalasalaku vacchannu)
2.	Wrong sentence ender	భాష గురించి నీకేమి తెలుసు (bhasha gurinchi neekemi thelsu)	It is interogative sentence and should be ended with a question mark (?)
3.	Error due to missing comma	నా షూస్ మెరువాలంటే పోలిష్ కీమత్ వుంది. (naa shoes meravalante polish theesukovali)	Coma is necessary after the dependent clause i.e., నా షూస్ మెరువాలంటే (naa shoes meravalante)
4.	Repeated/duplicate	నేను కళాశాలకు కళాశాలకు వెళ్ళాల్సి (nenu kalaselaku kalaselaku vellali)	Word కళాశాలకు (kalaselaku) has been typed twice

3.2. Errors the independent clause of compound sentences:

Under this category, errors related to mismatch of agreements between various word classes present in an independent

clause, order of modifier of noun phrase and order of words in verb phrase have been covered. Table 3.2 provides the list of errors covered under this category with suitable examples.

Table 3.2: Clause level (independent) errors covered in compound and complex sentences

1.	I Modifier and Noun agreement	1.1	Agreement in terms of Number	ఇద్దరు బాలుడు చూడబడి వెళ్ళారు. (iddaru baludu skooluki vellaru)
		1.2	Agreement in terms of Gender	అందమైన అమ్మాయిలు ఇంటి నుండి బయట బయటికి వెళ్ళింది. (andamaina ammayilu inti nundi bayatiki vellindi)
		1.3	Agreement in terms of Case	నలుపు అబ్బాయి అన్నం తిన్నాడు (nalupu abbayi annam thinnadu) నలుపు అబ్బాయిలు అన్నం తిన్నారు (nalupu abbayilu annam thinnadu) నలుపు అబ్బాయిలు అన్నం తింటారు. (nalupu abbayilu annam thintadu)
2.	Subject Verb agreement	2.1	Agreement in terms of Number	నలుగురు తమ్ముళ్ళు పని చేస్తున్నారు. (naluguru thammullu pani chesthunnadu)
		2.2	Agreement in terms of Gender	అమ్మాయి ఇండా వెగరు (ammayi jenda yeguru)
		2.3	Agreement in terms of Person	ఆ చూడబడి వెళ్ళుతారు (aa skool vellutharu) చరణ్ చూడబడి వెళ్ళింది (charan skoolki vellindi)
		2.4	Use of postposition with subject if the verb is transitive in perfect form	బాలుడు అన్నం తింటూ (baludu annam thinto)
3.	Order of word in Verb phrase	3.1		అబ్బాయి ఇంటిలో పోతున్నాడు (abbayi intilo pothunnadu)
				అబ్బాయి ఇంటికి వాస్తారు (abbayi intiki vastharu)
		3.2		అబ్బాయి ఇంటిలో వచ్చాడు (abbayi intilo vachadu) అబ్బాయి ఇంటి పోతాడు (abbayi inti pothadu)
		3.3		అబ్బాయి ఇంటి వచ్చారు (abbayi inti vacharu)

4. CONCLUSION:

This paper concerns the grammar checking of compound sentences in various agreement errors within independent clauses in case of compound sentences and between

dependent and independent clauses in case of compound sentences. This study will be helpful in identifying and separating the compound sentences from Telugu language. This study will be helpful in developing other Natural

Language Processing (NLP) applications like converting a compound sentence in simple sentences, grammar checking

of compound sentences.

5. REFERENCES:

1. Sanjeev kumar Sharma, G.S Lehel 'Identification of Compound Sentences in Punjabi Language' *Research Cell: An International Journal of Engineering Sciences, Inaugural Issue 2010* ISSN: 2229-6913 (Print), ISSN: 2320-0332 (Online) Vol. 1, pp. 1-8.
2. Beesley, K. R. 2001. Finite-state morphological analysis and generation of Arabic at Xerox Research: Status and plans in 2001. In *ACL Workshop on Arabic Language Processing: Status and Perspective Vol. 1*, pp. 1-8.
3. Bharati, A., Chaitanya, V., Sangal, R., & Ramakrishnamacharyulu, K. V. 1995. *Natural language processing: a Paninian perspective*. New Delhi: Prentice-Hall of India. pp. 65-106.
4. Bigert, J., Kann, V., Knutsson, O., & Sjobergh, J. 2004. Grammar checking for Swedish second language learners. pp. 33-47.
5. Bustamante, F. R., & Le6n, F. S. 1996. GramCheck: A grammar and style checker. In *Proceedings of the 16th conference on Computational linguistics-Volume 1*. Association for Computational Linguistics. pp. 175-181
6. Carlberger, J., Domeij, R., Kann, V., & Knutsson, O. 2004. The development and performance of a grammar checker for Swedish: A language engineering perspective. *Natural language engineering*, 1(1).
7. Chidambaram, D. 2005. Processing complex sentences for information extraction. A Thesis Presented in Partial Fulfillment of the Requirements for the Degree Master of Science.
8. Ehsan, N., & Faili, H. 2010. Towards grammar checker development for Persian language. *IEEE International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE)*, 2010. pp. 1-8
9. Fernandes, E. R., Pires, B., dos Santos, C. N., & Milidui, R. L. 2009. Clause identification using entropy guided transformation learning. *IEEE 2009 Seventh Brazilian Symposium in* Gill, M. S., Lehal, G. S., & Joshi, S. S. 2009. Part of speech tagging for grammar checking of Punjabi. *The Linguistic Journal*, 4(1), pp. 6-21.
10. *Information and Human Language Technology (STIL)*, pp. 117-124.
11. Hein, A. S. 1998. A Chart-Based Framework for Grammar Checking Initial Studies. In *Proc. of 11 th Nordic Conference in Computational Linguistic*. pp. 68-80.
12. Jurafsky, Daniel and James H. Martin. 2000. *Speech and Language Processing: An Introduction to Natural language Processing, Computational Linguistics, and Speech Recognition*. Pearson Education, Delhi, India
13. Kubon, V., & Platek, M. 1994. A grammar based approach to a grammar checking of free word order languages. In *Proceedings of the 15th conference on Computational linguistics-Volume 2*. Association for Computational Linguistics. pp. 906-910
14. Naber, D. 2003. A rule-based style and grammar checker. Thesis, Technical Faculty, University of Bielefeld, Germany
15. Narula, R., & Sharma, S. K. 2014. Identification and Separation of Simple, Compound and Complex Sentences in Punjabi Language. *International Journal of Computer Applications & Information Technology*. Vol. 6, Issue II Aug-September 2014.
16. Parveen, D., Sanyal, R., & Ansari, A. 2011. Clause Boundary Identification using Classifier and Clause Markers in Urdu Language. *Polibits Research Journal on Computer Science*, 43, pp. 61-65.
17. http://en.wikipedia.org/wiki/Telugu_language.
18. http://en.wikipedia.org/wiki/Telugu_grammar
19. http://simple.wikipedia.org/wiki/Telugu_language