Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.7702607

IDENTIFICATION OF SUBJECT - VERB AGREEMENT USING RULE-BASED APPROACH

Ankit Kumar Chouhan
Department of Computer Science & Engineering, Jabalpur Engineering College, Jabalpur
ankitkumarchouhan1998@gmail.com

Vishal Mani Mishra
Department of Computer Science & Engineering, Jabalpur Engineering College, Jabalpur
vishmishra8199@gmail.com

Dr. Jitendra Singh Thakur
Department of Computer Science & Engineering, Jabalpur Engineering College, Jabalpur
jsthakur@jecjabalpur.ac.in

Abstract
We see numerous attempts by the research community in the past few years to make the computer capable of understanding the Natural language of Humans. The advent of Machine learning & Deep learning gave a boost to this process and resulted in successful researches. However, these Deep learning models have their own downsides and still have some negative aspects. Recent developments in Subject - Verb Agreement research using LSTM models, RNN grammar etc. have shown good results but they lag as the complexity of the sentence increases. Another approach is given by Grammarly where they use a simple and efficient GEC sequence tagger using a Transformer encoder. Their system is pre-trained on synthetic data and then fine-tuned in two stages: first on errorful corpora, and second on a combination of errorful and error-free parallel corpora. In contrary to these approaches in this paper we have tried to focus on solving this problem using Rule based Methodology with the inspiration that deterministic methodologies are the best to solve deterministic problems. Using Rule based methodology to an extent where we have carefully defined rules in English Grammar helps us to generate good results with good accuracy of 81.5% F- Scores as follow F0.5:0.8311, F1:0.855, F2 : 0.88 and with lesser resources unlike used in learning models. However, these algorithms of rule-based approach can also be equipped with language models to solve problems which have ambiguity and really have a requirement of language modelling and learning models to increase the accuracy further and give improved results.
Keywords: Subject-Verb Agreement, Syntactical Error detection, Rule-based Approach

1 INTRODUCTION

As discussed earlier, there are several deep learning approaches proposed to identify Subject-Verb Agreement in the English Sentences. However, these methods have their own limitations. These methods have worked on several types of English sentences but still fail when the complexity of the sentences increases. The models use deep learning techniques which require large databases with heavy architecture to support the implementation of these techniques.

Recent approach proposed like using LSTM (Tal Linzen Et. al.) models and later modified using RNN models (Kuncoro et al.). These models learn syntax but only when they are big enough. Linzen et al. found that LSTM language models are not very good at predicting correct form, in cases when linear distance is unhelpful. Linzen et al. Found that the language model does okay on average, but it struggles on sentences in which there are nouns between the subject and verb. The language model does reasonably well (7% error) when there are no attractors, but this jumps to 33% error onsentences with one attractor, and a whopping 70% error (worse than chance!) on very challenging sentences with 4 attractors. Apart from this, Techniques developed for Full Grammar Correction (all types of Grammatical Errors) like Neural Machine Translation (NMT)-based approaches (Sennrich et al., 2016a) have become the preferred method for the task of Grammatical Error Correction (GEC)2. In this formulation, errorful sentences correspond to the source language, and error-free sentences correspond to the target language. Recently, Transformer-based (Vaswani et al., 2017) sequence-to-sequence (seq2seq) models have achieved state-of-the-art performance on standard GEC benchmarks (Bryant et al., 2019). Now the focus of research has shifted more towards generating synthetic data for pretraining the Transformer-NMT-based GEC systems (Grundkiewicz et al., 2019; Kiyono et al., 2019). NMT based GEC systems suffer from several issues which make them inconvenient for real world deployment: (i) slow inference speed,

(ii) demand for large amounts of training data and

(iii) interpretability and explainability; they require additional functionality to explain corrections, e.g., grammatical error type classification (Bryant et al., 2017) as mentioned in Omelianchuk et al., 2020. Omelianchuk et al. in GECToR-Grammatical Error Correction: Tag, Not Rewrite mention that they deal with the above issues by simplifying the task from sequence generation to sequence tagging.

Their GEC system consists of three training stages: pretraining on synthetic data, fine-tuning on an errorful parallel corpus, and finally, fine-tuning on a combination of errorful and error-free parallel corpora.

In contrast to this we propose a Rule-based approach where we have defined 40 rules of subject Verb agreement. Our motive of using the Rule-based approach is to leverage the capability of algorithmic programming rather than use the complex approaches of Artificial Intelligence and Deep Learning Models when we can use the traditional approach to solve the existing problems. We believe that the traditional approaches can be used to a greater extent so why not use them and switch to the Artificial Intelligence techniques only when we are sure that traditional approaches cannot solve them, as new Techniques require more advanced resources for processing and large amount of memory. Here we have specifically focused on Subject-Verb Agreement in English Sentences only. Our approach works by finding syntactic dependencies between the words and the POS tags of the words in the English Sentences. The Dependencies and POS tags are found using the Stanford Parser. In our work we found that English Sentences have unique combination of syntactic dependencies and POS tags, and this feature helps us to classify sentences into different groups. If a particular set of dependencies are found in the given sentence this confirms that the sentence belongs to a particular case and hence it satisfies the Subject Verb agreement in the sentence. The procedure is applied in a systematic manner for sentences containing countable nouns as well as collective nouns and other simple sentences as well. Since this is a Rule-Based approach we do not have any Training set but we have tested our Algorithm on several datasets. Overall, we found the approach yields an accuracy of 81.5% when applied to a set of English sentences from a dataset of CoNLL 2014 test set (Ng et al., 2014).

2 BACKGROUND: SUBJECT – VERB AGREEMENT

According to a definition from Cambridge dictionary, the person and number of the subject of the clause determine the person and number of the verb of the clause. This is called subject–verb agreement or concord. Agreement or concord happens when a word changes form depending on the other words to which it relates. In Standard English, one may say I am, or he is, but not "I is," or "he am". This is because the grammar of the language requires that the verb and its subject agree in person. The pronouns I and he are first, and third person respectively, as are the verb forms am and is. The verb form must be selected so that it has the same person as the subject in contrast to notional agreement, which is based on meaning. Based on English grammar rules the verb form for English third person present tense depends on whether the syntactic subject of the sentence is plural or singular. For example, the sentences given below are different combinations of subject and verb in number. However, only sentences (i) and (iii) are correct. Sentences marked in * are unacceptable sentences.

i. The book is on the shelf
*ii. The book are on the shelf.
iii. The books are on the shelf.
*iv. The books is on the shelf.

While in these examples the subject's head is adjacent to the verb, in general the two can be separated by some sentential material also as follows:

(1) The keys to the cabinet are on the table.

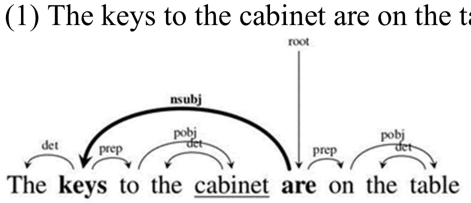


Fig. 1

Note: This example is taken from Tal Linzen, Emmanuel Dupoux -- Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies

If we carry out a syntactic parse of the sentence, we can easily identify the head of the nsubj arc that corresponds to the verb and we can easily find out the number of the verb. The form of the verb is identified by the head of subject which is directly connected to it via a nsubj arc. Other nouns which are present between the subject and the corresponding verb need to be ignored. This is an important class of sentences as this increases the complexity of sentences and is a major issue in increasing the error rate of deep learning models used to identify subject

– verb agreement in sentences and a reason we propose a rule based approachto solve these issues.

3 RELATED WORK

There have been numerous researches in the last few years working on subject verb agreement issue either specifically on this topic or as a part of a larger problem. Related work by Chung-Huang , Mei Hua chen, et. al(2011) in EdIt: A Broad- Coverage Grammar Checker

Using Pattern Grammar introduced a method for detecting grammatical errors in learner's writings and provide suggestions. This method's work involves parsing a reference corpus and inferring grammar patterns in the form of a sequence of content words, function words, and parts-of-speech.

Then, at run-time the given passage by user is analysed by the module and matched with the inferred grammar patterns obtained from the reference corpus. This match is carried out using an algorithms named Levenstein algorithm to detect errors in the input if present and provide suggestions. However in this approach promising results are obtained in three common types of errors. This approach examined three types of errors and a mixture of them namely Incorrect word form, Incorrect preposition, and solving confusion between intransitive and transitive verb. Another work mentioned in Automatic Grammar Correction for Second-Language Learners (2006) by John Lee, Stephanie Seneff describe an approach to grammar correction by generating a word lattice of candidate words in an incorrect input. In this approach traditional n - gram model is used to generate a set of n- candidates having an accuracy ranging from 80 %- 90 %. One of the most recent works in this problem statement are from Tal Linzen, Emmanuel Dupoux, Yoav Goldberg (2016) in Assessing the ability of LSTMs to learn Syntax-Sensitive dependencies. As mentioned in the name itself the motive of Linzen et al was to assess the capability of LSTMs to learn Syntax sensitive dependencies. Linzen et al used the LSTM (long short- t e r m memory) artificial recurrent neural network architecture for this purpose which are basically used in as deep learning models. It was found that LSTM language models are not very good at predicting correct form , in cases when linear distance is unhelpful. It was found that the language model does okay on average , but it struggles on sentences in which there are nouns between the subject and verb. With strongly supervised settings the LSTMs achieved very high overall accuracy (less than 1% errors) but errors increased when sequential and structural information increased. As mentioned in a blog by Stanford NLP group the error rate increased in the LSTM model as the number of attractors increased in the sentences. (Intervening nouns with the opposite number from the subject are called agreement attractors.) In the case of 4 attractors the error rate even increased to about 70%. In a development to Linzen's approach Kuncoro et al, (2018) in LSTMs can learn Syntax-sensitive dependencies well, but modelling structure makes them better showed that there can be an improvement to the approach of Linzen and there work showed that error rates decreased to a greater extent and at 4 attractors only 17.6 % error rate by increasing the network capacity. Contrary to the findings of Linzen et al., Kuncuro's experiments suggest that sequential LSTMs are able to capture structural dependencies to a large extent, even for cases with multiple attractors. His finding suggests that network capacity plays a crucial role in capturing structural dependencies with multiple attractors. He finds out that RNN outperform sequential LSTM model for cases with multiple attractors. Another technique is given by Omelianchuk et al., 2020 of Grammarly which is a generalized approach considering all types of errors in English sentences. We find all these methods and approaches use machine learning or deep learning models in some or the other way. However, contrary to this we propose an approach where we use complete Rule based approach to solve the task.

4 OUR PROPOSED APPROACH

We have seen how LSTM model [Linzen et al.] and their sequential approach [Adhiguna kuncoro] work their best in finding Subject Verb Agreement. Contrary to this model, our method suggest that we can go with the rule based approach in order to increase efficiency in finding the errors in subject verb agreement. More complex structural dependencies can be easily determined if we use a proper well defined rules over them. Our experience suggests that if we find a well-defined rule by finding mapping over dependencies and pos- tags, we get a highly accurate model over the given set. Even Multiple attractors cannot escape the rule defined on them. We find that a strong character LSTM language model or sequential LSTM model performs much worse in the number agreement task. The main question arise here is that how our model work better than existing models. Learning models are good but our model is defined doing proper research on how similar sentence structure show same relation over dependencies and POS-tags. Using these mapping we found out 40 rules to categorize sentence set as of now. (Rules are enlisted in Appendix 1) We created our model in Python Programming Platform—Python is an interpreted, high-level and general-purpose programming language which is highly user- friendly language —with the help of Stanford parser. Object Oriented feature of Python helped a lot in designing our model.

5 DATASETS

For Creating Rules for our algorithm we used our own created dataset which consisted of sentences obtained from Different sources having around 200 correct sentences. For Testing purpose we used 2 datasets : first was the CoNLL -2014 test set through which we obtained the incorrect sentences and also the correct sentences. The incorrect sentences did not consist of any other type of error other than Subject-Verb Agreement, second dataset was obtained from Kaggle.

Dataset	Type	# sentences
Self-Created	Training	#Correct: 141
		#Incorrect:141
Co-NLL	Test	#Correct: 356
		#Incorrect:179

Our Proposed Approach works as follows:

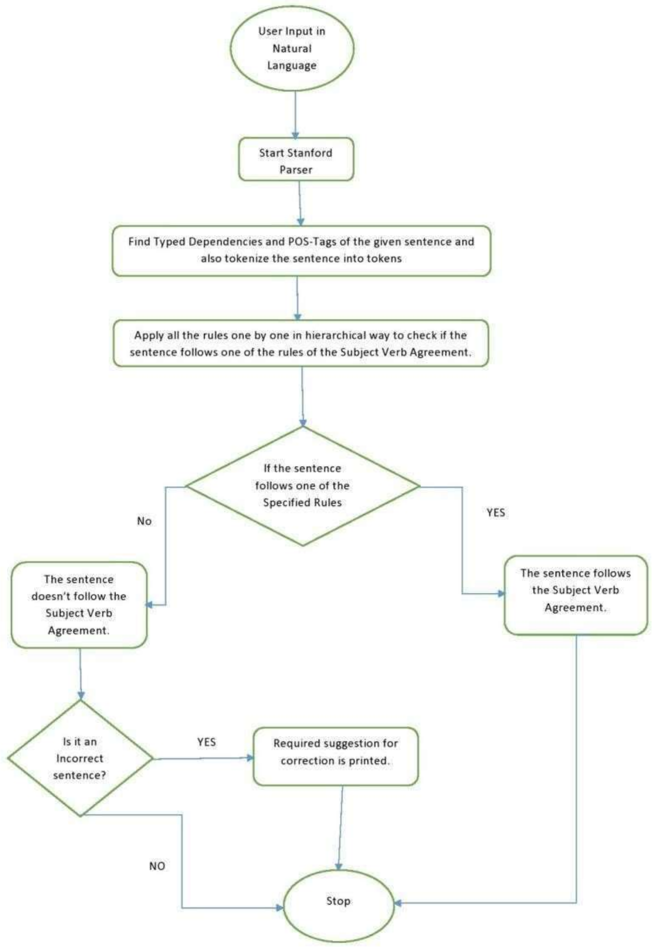
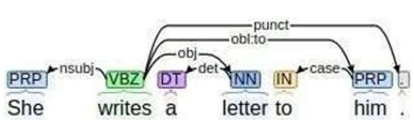
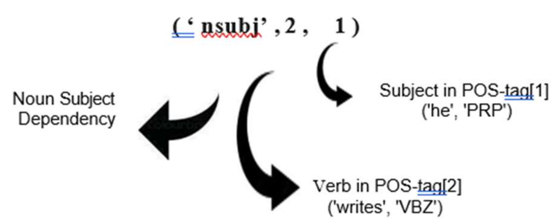


Fig. 2 Flow diagram of the process

Let's take some examples to understand it better the process.

Example 1: **"She writes a letter to him.**





In list format the mapping can be stated as:

- The respective pos-tag is [('She', 'PRP'), ('writes', 'VBZ'), ('a', 'DT'), ('letter', 'NN'), ('to', 'TO'), ('him', 'PRP')]
- The respective dependencies are:- [('ROOT', 0, 2), ('nsubj', 2, 1), ('det', 4, 3), ('dobj', 2, 4), ('case', 6, 5), ('nmod', 4, 6)]

In this case we can observe dependency ('nsubj', 2, 1) , gives relation between noun subject and the verb. In this case subject is singular 'She' and here verb form is 'VBZ' which always maps to singular subject. Hence output here is: - "The rule is S-S subject verb agreement > subject = he; Verb = writes"

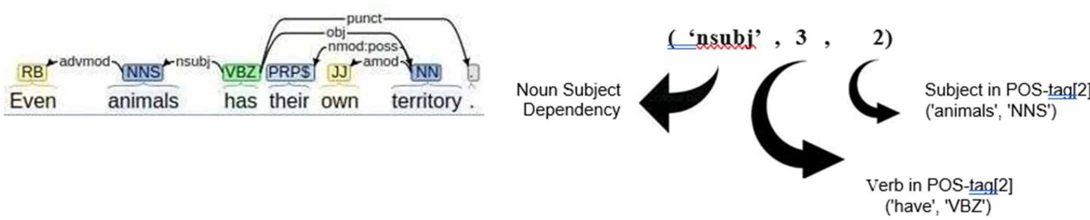
Example 2: "Even animals have their own territory."

***Example 2:* "Even animals have their own territory."**

- The POS-tags are [('even', 'RB'), ('animals', 'NNS'), ('have', 'VBP'), ('their', 'PRP$'), ('own', 'JJ'), ('territory', 'NN'), ('.', '.')].
- And the respective dependencies are [('ROOT', 0, 3), ('advmod', 2, 1), ('nsubj', 3, 2), ('nmod:poss', 6, 4), ('amod', 6, 5), ('dobj', 3, 6), ('punct', 3, 7)]

In this case we have plural subject i.e. animals and hence it is getting mapped to plural verb form "have" having pos-tag VBP(plural verb). The rule is plural subject takes plural verb [R2]. Output:- " The rule is P-P subject verb agreement --- > subject = animals Verb = have."

Example 3 : *- In this example we are taking an incorrect sentence to demonstrate what kind of output will be displayed in the wrong scenario.* **"Even animals has their own territory."**



- The POS-tags are [('even', 'RB'), ('animals', 'NNS'), ('has', 'VBZ'), ('their', 'PRP$'), ('own', 'JJ'), ('territory', 'NN'), ('.', '.')]
- And the respective dependencies are [('ROOT', 0, 3), ('advmod', 2, 1), ('nsubj', 3, 2), ('nmod:poss', 6, 4), ('amod', 6, 5), ('dobj', 3, 6), ('punct', 3, 7)]

Here we get singular verb ('VBZ') for plural subject ('NNS'). Hence it is a wrong sentence and an error message with suggestion will be displayed as output in this case.

6 RESULT

There are three possible cases that can occurs after passing input to our model:-

- Sentence gets detected correctly in our model If it is a grammatically correct sentence.
- Grammatically Wrong SVA sentences are getting detected.
- Since the model needs to be extended with more rules to cover more sentences, some sentences goes undetected because if they are grammatically incorrect. In that case we are considering those sentences as incorrect.
- Some correct sentences also goes undetected which we are not considering for accuracy detection. (Precision and recall)

7 DISCUSSION AND ANALYSIS

Using train set of 282 distinct sentence structure, we build our rule-based model. We are comparing our model with Grammarly. Subject verb agreement from CoNLL -2014 test dataset is used as standard dataset. 535 Incorrect and Correct sentences were run over our model. These sentences were obtained from the dataset using the annotations given in the dataset along with the sentences, and following is the analysis:-

➢ Total input sentences = 535

➢ Total Actual correct sentences = 356

➢ Total Actual Incorrect sentences = 179

➢ Number of undetected correct sentences = 106(Not considered in analysis because of model incompleteness)

➢ Number of undetected incorrect sentences = 56(These are considered as incorrect sentences since model is trained on correct sentences)

	Our Model	Grammarly
Accuracy	81.5%	85.57%
Precision	0.816	0.9355
Recall	0.898	0.86
(F_Score with Beta 0.5) $F_{0.5}$	0.8311	0.9188
(F_Score with Beta 0.5) F_1	0.855	0.891
(F_Score with Beta 0.5) F_2	**0.88**	0.87
	TP= 204 FP = 46	TP= 334 FP = 23
	FN= 23 TN= 100	FN= 51 TN= 105

Note: The F-beta score is a weighted harmonic mean between precision and recall, and is used to weight precision and recall differently. It is likely that one would care more about weighting precision over recall,

which can be done with a lower beta between 0 and 1

8 CONCLUSION

Despite Grammarly model is working sufficiently fine on determining subject verb agreement, we believe that since our model is not yet completed and more rules can be added to it , it is giving a good performance without heavy resources required. Our Rule Based Model can be improved in future so that it will give exponentially good result. As of now, our model can compete with the Grammarly model and even give better F-Beta score in some cases. The rule-based model can be integrated with the Grammarly model to create a big scope of sentences to be covered in all. On other hand, our model work better with less complexity and few rules. Time complexity is still a bigger concern in all the existing models. In future, many other rules can be added in our existing rules or our rules can be further being improved to accurately detect large range of sentence structure set with much less error rates and cover maximum possible structures. We can improve this model till that extent, so we reach to maximum possible accuracy and works more on reducing the time complexity.

9 REFERENCES

a. High School English Grammar and Composition, P. C. Wren and H. Martin[Book]

b. How to write Correct English , R.P Singh[Book]

c. Subject-Verb Agreement A “Grammar Help Worksheet” by Abbie Potter Henry

d. Assessing the Ability of LSTMs to Learn Syntax-Sensitive Dependencies ,Tal Linzen, EmmanuelDupoux,Yoav Goldberg [2016]

e. LSTMs Can Learn Syntax-Sensitive Dependencies Well But Modeling Structure Makes ThemBetter, Adhiguna Kuncoro, Chris Dyer, John Hale, Dani Yogatama, Stephen Clark, Phil Blunsom[2018]

f. Targeted Syntactic Evaluation of Language Models, Rebecca Marvin, Tal Linzen[2018]

g. Stanford typed dependencies manual Marie-Catherine de Marneffe and Christopher D.Manning ,September 2008 ,Revised September 2016

h. A Study of Subject-Verb Agreement: From Novice Writers to Expert Writers, Surina NayanKamaruzaman Jusoff

i. Automatic Grammar Correction for Second-Language Learners, John Lee, Stephanie Seneff

j. EdIt: A Broad-Coverage Grammar Checker Using Pattern Grammar Chung-Chi Huang Mei-HuaChen, Shih-Ting Huang Jason S. Chang

k. Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Edinburgh neural machine translation systems for WMT 16. In Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers, pages 371–376, Berlin, Germany. Association for Computational Linguistics.

l. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Advances in neural information processing systems, pages 5998–6008.

m. Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. 2019. The BEA-2019 shared task on grammatical error correction. In Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 52–75, Florence, Italy. Association for Computational Linguistics.

n. Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Kenneth Heafield. 2019. Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications, pages 252–263.

o. Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. Automatic annotation and evaluation of error types for grammatical error correction. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 793–805, Vancouver, Canada. Association for Computational Linguistics.

p. Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. 2014. The conll-2014 shared task on grammatical error correction. In Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task, pages 1–14

q. Common Grammatical Errors: Subject-Verb Disagreement,Bethune College, NY University, [https://bethune.yorku.ca/writing/s_v/]

r. Subject Verb Agreement Rules [https://www.grammarbook.com/grammar/subjectVerbAgree.asp]

s. A Light Rule-based Approach to English Subject-Verb Agreement Errors on the Third PersonSingular Forms Yuzhu Wang, Hai Zhao, and Dan Shi
t. These 27 Rules Of Subject Verb Agreement, https://akashgautam.com/blog/2011/11/25/english-grammar-rules-subject-verb- agreement/

10 APPENDIX 1

Table 1: Rules to identify Subject Verb Agreement

Rule #	Rule	Condition to be satisfied in sentence
R1	Simple Past	cop(A,B), POS= 'VBD' & VBD_keyword = 'were'
R2	Singular Nouns with plural verbs	nsubj(A,B), cop(C,D) , verb = 'are' or 'were'
R3	Present tense with present participle	nsubj(A,B), aux(C,D) , if POS = 'VBP' & VBP_index-1 = 'NNS' & VBP_index+1 = 'VBG'
R4	Present tense with past participle	nsubj(A,B), aux(C,D) , if POS = 'VBP' & VBP_index-1 = 'NNS' & VBP_index+1 = 'VBN'
R5	Past tense with present participle	nsubj(A,B), aux(C,D) , if POS = 'VBD' & VBD_index-1 = 'NNS' & VBD_index+1 = 'VBG'
R6	Past tense with past participle	nsubj(A,B), aux(C,D) , if POS = 'VBD' & VBD_index-1 = 'NNS' & VBD_index+1 = 'VBN'
R7	Keywords: 'The number ' in present tense	Word = 'The' ; word + 1 = 'number' ; POS tag in POS list = 'VBZ'
R8	Keywords :'The number ' in past tense	Word = 'The' ; word + 1 = 'number' ; POS tag in POS list = 'VBD' ; POS keyword = 'was'
R9	Keywords:'A number ' in present tense	Word = 'The' ; word + 1 = 'number' ; POS tag in POS list = 'VBP'
R10	Keywords :'A number ' in past tense	Word = 'A' ; word + 1 = 'number' ; POS tag in POS list = 'VBD' ; POS keyword = 'were'
R11	Keywords: 'each', 'every', 'either', 'neither', 'everyone', 'many a' etc. In present tense	cc:preconj(A,B) ; POS = 'VBZ'
R12	Keywords: 'each', 'every', 'either', 'neither', 'everyone', 'many a' etc. In past tense	cc:preconj(A,B) ; POS = 'VBD'; POS keyword = 'was'
R13	Keywords : anybody, everybody, nobody, somebody, anything, something, everything etc in present tense	if 'anybody' or'everybody' or 'nobody' or 'somebody' or 'anything' or 'everything' or 'something' or 'nothing' or 'anyone' or 'everyone' or 'someone' in sentence and POS = 'VBZ'
R14	Keywords : anybody, everybody, nobody, somebody, anything, something, everything etc in future tense	if 'anybody' or'everybody' or 'nobody' or 'somebody' or 'anything' or 'everything' or 'something' or 'nothing' or 'anyone' or 'everyone' or 'someone' in sentence and ;POS = 'VB'; POS['VB'-1]= 'MD' & POS['VB'-1][0] = 'will' or 'shall'
R15	Keyword: 'No one'	If 'no one ' in sentence; POS = 'VBZ'
R16	Keyword 'Only one of..' in present tense	If 'Only one of' in sentence; cop(A,B) ; POS[index(cop)+2-1][1] = 'VBZ'
R17	Keyword 'Only one of..' in past tense	If 'Only one of' in sentence; cop(A,B) ; POS[index(cop)+2-1][1] = 'VBD' & POS[index(cop)+2-1][0] = 'was'
R18	Keyword 'one of those' in present tense	If 'one of those' in sentence; cop(A,B) ; POS[index(cop)+2-1][1] = 'VBP'
R19	Keyword 'one of those' in past tense	If 'one of those' in sentence; cop(A,B) ; POS[index(cop)+2-1][1] = 'VBD' & POS[index(cop)+2-1][0] = 'were'
R20	Keyword : 'One of the things' in present tense	If 'one of the things' in sentence; cop(A,B) ; POS[index(cop)+2-1][1] = 'VBP'
R21	Keyword : 'One of the things' in past tense	If 'one of the things' in sentence; cop(A,B) ; POS[index(cop)+2-1][1] = 'VBD' & POS[index(cop)+2-1][0] = 'were'
R22	Keyword :'One of the'or 'One of' in present tense	If 'one of the' in sentence; cop(A,B) or aux(A,B) ; POS[index(cop)+2-1][1] = 'VBZ'
R23	Keyword :'One of the'or 'One of' in past tense	If 'one of the' in sentence; cop(A,B) or aux(A,B) ; POS[index(cop)+2-1][1] = 'VBD' & POS[index(cop)+2-1][0] = 'was'
R24	Keyword: 'of' in present tense sentence	nmod(A,B); if A=='of' ;POS= 'NN' & 'NNS' not in POS list; if 'VBZ' in POS list & index('VBZ')>index(A)
R25	Keyword: 'of' in past tense sentence	nmod(A,B); if A=='of' ;POS= 'NN' & 'NNS' not in POS list; if 'VBD' in POS list & index('VBZ')>index(A) & keyword('VBD')=='was'
R26	Keyword : 'of' with collective nouns like 'team', 'set', chain, bunch, bouquet, merit,class, galaxy, series, batch , band, herd' in present tense	If 'NNS' & 'VBP' in POS list, index('VBP ') > index('NNS')

R27	Keyword : ‘of’ with collective nouns like ‘team’, ‘set’, chain, bunch, bouquet, merit,class, galaxy, series, batch , band, herd’ in past tense	If ‘NNS’ & ‘VBD’ in POS list, index(‘VBP ’) > index(‘NNS’) & keyword(‘VBD’) == ‘were
Rule #	Rule	Condition to be satisfied in sentence
R28	Plural subjects take plural verbs.	(nsubj,A,B), NNS, NNPS, LS, FW, VBP, VBN, VD
R29	Singular subjects take singular verbs.	(nsubj,A,B), NN, NNP, VBZ, VB, VBD
R30	Dare not/ Need not (s-p Exception)	Keyword :- dare not, need not
R31	Noun+Preposition+N+P+N+V	(nsubjpass,A,B)
R32	Subject joined by 'AND' are usually plural and uses plural verb.(When two subjects are not in the sense of one)	(nsubjpass,A,B)or (nsubj,A,B) AND (aux,E,F) or (auxpass, E,F) or (Cop, E,F)
R33	Subject joined by 'AND' which sense to single person, thing etc, uses singular verb.	(nsubjpass,A,B) OR (nsubj,A,B), (aux,E,F)
R34	Rule when both singular and plural subjects are present. Verb agrees with the nearer subject.	(Nsubj,A,B), (conj,C,D), (aux,E,F)
R35	Plural verbs are required for many nouns that have no singular form, such as proceeds, goods etc.	Keyword:- (nsubj,A,B), NNS, NNPS
R36	When nouns expressing periods of time, amounts of money, numbers are considered as a singular unit, singular verb.	(Numode,A,B), (nsubj,C,D), Keyword:- ‘number’ given in the sentence.