Grammar Correction using Rule Based System

Requirement of project:

A user would provide a natural language query which would be used as input by the program and the expected output would be to return the grammar corrected sentence.

Example case:

User input:

This is a NLP relatad task

Expected output:

This is an NLP related task

Explanation:

In the above sentence ‘relatad’ is replaced by ‘related’ and ‘a’ is replace by ‘an’ which would be the required output for grammar corrector to make the user queried sentence grammatically correct.

How it benefits natural language understanding:

It leads to better sentence parsing which creates better dependency tree and ultimately helps in better information extraction

Implementation Structure –

Phase 1:

Flow of data for Grammar Correction:

1.Spelling Checker (To be implemented by - Aviral Mishra)

INPUT-> ‘This is a NLP relatad task’

OUTPUT-> ‘This is a NLP related task’

2.Word Tokenization (To be implemented by - Aviral Mishra)

INPUT-> ‘This is a NLP related task’

OUTPUT-> ‘This’ , ‘is’ , ‘a’ , ‘NLP’ , ‘related’ , ‘task’

3.Part-of-speech tagging (To be implemented by - Chakori Chaturvedi)

INPUT-> ‘This’ , ‘is’ , ‘a’ , ‘NLP’ , ‘related’ , ‘task’

OUTPUT-> (‘This’,’DET’) , (‘is’,’AUX’) , (‘a’, ‘PROPN’) , (‘NLP’,’NOUN’) , (‘related’,’VERB’) ,(‘task’, ‘NOUN’)

4.Chunking Phrases (To Be implemented by - Sai Maniteja Penugonda)

INPUT -> (‘This’,’DET’) , (‘is’,’AUX’) , (‘a’, ‘PROPN’) , (‘NLP’,’NOUN’) , (‘related’,’VERB’) ,(‘task’, ‘NOUN’)

OUTPUT-> (‘This’,’DET’) , (‘is’,’AUX’) , (‘a’,’PROPN’) , (‘NLP related task’,’NOUN’)

5.Rule Matching (Minimum 3 rules to be implemented by each Aviral Mishra, Chakori Chaturvedi, Sai Maniteja Penugonda)

EXAMPLE RULE:

rule of replacing a to an or vice versa according to the structure of sentence

INPUT-> ('This','DET') , ('is','AUX') , ('a','PROPN') , ('NLP related task','NOUN')

OUTPUT-> 'This' , 'is' , 'an' , 'NLP related task'

6.Modifying query with the output of previous state (To be implemented by all)

Integrating the output of all the rules to create the final grammatically correct query

Final OUTPUT -> 'This is an NLP related task'

References:

Spelling Checker-

Many open-source libraries like pyspellchecker are available please try to integrate this or a similar library

Word Tokenization, Part of speech tagging and chunking phrases-

Spacy library could be used for the above requirement

For reference you could use https://spacy.io/api/token

Rule Matching-

Implement the existing rules that are already known

For reference you could understand it here

http://www.danielnaber.de/languagetool/download/style_and_grammar_checker.pdf

feel free to use other resources available which provide open-source data that can be loaded in project and used for rule matching or any other requirement.

Phase 2:

1. Implementing more rules to increase the accuracy of the grammar correction project (To be implemented by all)
2. Creating new rules that are still not available in open-source grammar correction datasets

   How rules are made:

   1. Pick an undetected grammar error
   2. Use the error word and the words from its context to make a rule, e.g. were are -> we are
   3. Generalize the rule by part of speech tagging instead of words.
   4. Check if the rule also doesn't exist in correct sentences by using an open-source dataset.

3. Packaging the project: (To be implemented by all)

   1. A function that user could integrate in their project and call that function to directly get the grammar corrected sentence.

   2. Making the project available as a pip installable package

      EXPECTED steps to implement in other projects

      1. Installing the package -> pip install <package>
      2. Import the function to grammatically correct sentence -> from <package_name> import <grammar_corrector>
      3. Using it to grammatically correct any user query -> grammar_corrector("This is a NLP relatad task")