A PROJECT REPORT ON

अन्वयार्थ (Anvayartha)
Paraphrasing Tool for Low Resource Language

SUBMITTED TO THE
CUMMINS COLLEGE OF ENGINEERING FOR WOMEN, KARVENAGAR, PUNE
(an autonomous institute affiliated to savitribai phule pune university.),
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF TECHNOLOGY (COMPUTER ENGINEERING)

SUBMITTED BY

STUDENT NAME	CNUM	ROLL NO
Pranita Barbade	C22019881908	4908
Akshada Malpure	C22019881934	4934
Anushka Pawar	C22019881945	4945
Reena Prasad	C22019881946	4946

CERTIFICATE

This is to certify that the project report entitles

अन्वयार्थ (Anvayartha)
Paraphrasing Tool for Low Resource Language

Submitted by

STUDENT NAME	CNUM	ROLL NO
Pranita Barbade	**C22019881908**	**4908**
Akshada Malpure	**C22019881934**	**4934**
Anushka Pawar	**C22019881945**	**4945**
Reena Prasad	**C22019881946**	**4946**

is a bonafide student of this institute and the work has been carried out by her under the supervision of **Prof. Pranjali Deshpande** and it is approved for the partial fulfillment of the requirement of Cummins college of engineering for women, karvenagar, pune (an autonomous institute affiliated to savitribai phule pune university.), for the award of the degree of **Bachelor of Technology** (Computer Engineering).

(Prof. Pranjali Deshpande)
Guide
Department of Computer Engineering

(Dr. Supriya Kelkar)
Head,
Department of Computer Engineering

(Dr. M. B. Khambete)
Principal,
Cummins College of Engineering for Women Pune – 52

Place : Pune

ACKNOWLEDGEMENT

Our profound gratitude goes to our wonderful guide, **Prof. Pranjali Deshpande** for her invaluable support, patience, time and guidance in seeing us to the completion of this research work. Also our gratitude goes to **Dr. Supriya Kelkar** , Head , Department of Computer Engineering for top of the state labs for project work.

We also extend gratitude and appreciation to **Dr. M. B. Khambete ,** Principal, Cummins College of Engineering for Women Pune .

We also wish to acknowledge the great support of our parents, siblings who have been a source of inspiration towards our academic pursuit.

Pranita Barbade
Akshada Malpure
Anushka Pawar
Reena Prasad

ABSTRACT

Nowadays, most of the automatic work has been done in English language but not much work has been done in low resource language. The main objective is to convert the existing sentence in different form by remaining the semantic or meaning same. This will helpful in converting the complex sentence into simpler one.
Methods/ Statistical Analysis: Our system mainly deals with low resource language Sentences and its different forms.

TABLE OF CONTENTS

01.INTRODUCTION

Natural Language Processing (NLP) consists of two concepts: Natural Language Understanding (NLU) and Natural Language Generation (NLG). NLU helps the machine to understand the data, it is used to interpret data to understand the meaning of the data to be processed accordingly. It solves it by understanding the context, semantic, syntax, intent, and sentiment of the text. NLG is a process to produce meaningful sentences in Natural Language. It explains the structured data in a manner that is easy to understand for humans with a high speed of thousands of pages per second. Some of the NLG models are Markov chain, Recurrent neural network (RNN), Long short-term memory (LSTM), and Transformer. NLG innately requires NLU as well! NLG is quite a less explored part of NLP but has a wider area of applications like in Weather Reports, Image Caption, Chatbots, Voice assistants, AI blog writers Paraphrasing etc.

Paraphrasing means to express something again using different words. Paraphrase sentences have always been a useful mechanism in various academic applications. It helps not only to keep a check on plagiarism but serves many other purposes which include increasing the clarity of a given sentence by presenting it in other words.

Paraphrasing can be divided into two levels:

1) Keyword Replacement.
2) Sentence Regeneration.

In the first level, only the keywords (the words with the most significance) in the sentence are replaced without changing the meaning of the sentence. Here, the new sentence generated is not entirely different from the original sentence. In the second level, the sentence is paraphrased into a completely new sentence. Here, based on the contextual information of the sentence, a new sentence is generated. The newly generated sentence is completely different from the original sentence.

But due to various challenges, from Pragmatics to Discourse, understanding any natural language is a difficult task for a machine. This task becomes more rigorous in the case of low-resource languages. Low-resource languages are the ones that have little to no internet presence. Many Indo-Aryan languages such as Marathi, Hindi, and Punjabi which have originated from Sanskrit are low-resource languages. These languages lack large mono-lingual or parallel corpora. This makes it extra challenging in developing NLP applications based on a low-resource language.

Therefore, there is a need to create a tool for paraphrasing low resource language (Marathi) sentences as it will solve problems such as:

1) Making it easier to understand the meaning of Marathi sentences.
2) Help in widespread use of Marathi Literature.
3) Help new learners of the Marathi language to understand the sentences.

1.1 OVERVIEW & MOTIVATION

Text regeneration is the process of reforming a provided input sentence into a desired output sentence. Paraphrasing of sentences is one such process under text generation. Paraphrasers has multiple application in translators and chatbots as well as text paraphrasing bots. There are multiple models such as T5 and GPT which are developed for text regeneration and are used all over the industry. Libraries such as key2text and parrot which are developed in python can be readily imported and used for paraphrasing. However, the drawback of these models is that all of them are developed around English as the input and output language. They fail to provide results when applied on any low resource language.

```
Input_phrase: Can you recommed some upscale restaurants in Newyork?
--------------------------------------------------------------------------
list some excellent restaurants to visit in new york city?
what upscale restaurants do you recommend in new york?
i want to try some upscale restaurants in new york?
recommend some upscale restaurants in newyork?
can you recommend some high end restaurants in newyork?
can you recommend some upscale restaurants in new york?
can you recommend some upscale restaurants in newyork?
--------------------------------------------------------------------------
Input_phrase: What are the famous places we should not miss in Russia
--------------------------------------------------------------------------
what should we not miss when visiting russia?
recommend some of the best places to visit in russia?
list some of the best places to visit in russia?
can you list the top places to visit in russia?
show the places that we should not miss in russia?
list some famous places which we should not miss in russia?
```

Fig 1.1 Paraphrased English Sentences

The above image shows input and output of an English paraphrase tool. The input sentences are passed to paraphraser which generates multiple paraphrased sentences and displays them as output. Availability of such a tool for other low resource language is still a concern. This provides a motivation for more research and work to be done in the development of one such paraphrasing tool for other low research language. Development of such a tool will not be beneficial for the language but will also act as a learning assistance for the new learners of the language.

The aim of this research is to develop a parrot like tool for some other Indian language. Indian Institute of Technology Bombay had developed wordnet for various Indian languages. This wordnet can act as an assistance in the development of dataset for the training of the model.

However, currently the need is to developed such a model from scratch for the language chosen for implementation.

1.2 PROBLEM DEFINITION AND OBJECTIVES

Problem Definition:
To create a tool for paraphrasing of a low resource language.
Understanding of a low resource language is still a non-trivial task. This is the direct outcome of inadequate amount of research done in this domain. The tool to be developed would be used for text regeneration (paraphrasing) which may prove beneficial in many NLP applications.

The model to be used in the tool would require the user to provide a grammatically and syntactically correct sentence in the language implemented. Later the input would be passed to the model which will generate multiple paraphrased sentences as the output.

The initial approach would be to use the synonyms, antonyms, idioms as a replacement to a keyword in the sentence. The keyword can be directly replaced by its synonym and not of its antonym. The model may check for the presence of an idiom in the sentence and if present may change it to its literal sentence. This word could later be extended to paraphrase the entire sentence.

Objectives:

Broader Objective:

To create a tool for paraphrasing of low resource language sentence.

Sub-Objectives:

- Creating dataset and Data Pre-processing.
- Identification of candidate keywords.
- Replace the candidate keywords using wordnet.
- Evaluation of the multiple paraphrased sentences

1.3 PROJECT SCOPE AND LIMITATIONS

Scope:
Text Regeneration:

- Generation of input dataset.
- Accessing Wordnet.
- Implement various algorithms to identify candidate keywords for paraphrasing.
- Evaluating the paraphrased sentences using similarity index.

Limitation:

- Input sentences in language other than the one implemented.
- Translation from other languages into the language implemented in the model.
- Improvement of model in understanding the semantics (NLU).

02.LITERATURE SURVEY

2.1 BACKGROUND OF DOMAIN

In paper [1], A novel approach with two discriminators and multiple generators to generate a variety of different paraphrases. Significant increase in diversity. Improvement in generation quality. In paper [1], To apply our framework to increase reference texts for automatic evaluation or augment training data for text classification.

The paper [2], talks about converting an idiomatic sentence into it's literal sentence. It has used LSTM seq2seq transformer for the same. In paper [2], the model follows simple pattern matching and paraphrases only idiomatic part of the
sentence. The model may not work for metaphorical sentences.

In paper [3].the model produces another sentence without changing its semantic after applying synonyms and antonyms replacement method. Wampserver is used as database. The aim was to make a decision support system that will work in similar manner like humans and can respond just like humans by understanding the different forms of sentences given to it as
an input.

In Paper [4], paper proposes a method that is combination of deep generative models (vae) with sequence-to-sequence models (lstm) to generate paraphrases. It uses a variational autoencoder (vae) as a generative model. In paper [4], module used is quite complex and not modular and exhaustive training of dataset is required.

In paper [5], This paper propose a deep generative model to generate paraphrase with diversity. This model is based on an encoder-decoder architecture.
In paper [5], To apply our framework to increase reference texts for automatic evaluation or augment training data for text classification.

2.2 COMPARISONS OF RESEARCH PAPERS STUDIED

Paper	Approaches and Techniques Used	Remarks
Indian Journal of Science and Technology, 2016 [2]	The input paragraph is first broken down into sentences, followed by tokenization of sentences into words and appropriate reframing rules are applied for paraphrasing.	• Paraphrasing techniques like replacement by synonym or antonym used.
Sprenger, 2021 [3]	LSTM seq2seq transformer converts idiomatic sentence into literal sentence.	• Simple pattern matching instead of complete paraphrasing. • Incapable for metaphorical sentences.
Association for Computational linguistics, 2019 [4]	Approach with discriminators and multiple generators to generate a variety of different paraphrases.	• Model has better results concerning both diversity and quality
arXiv, 2019 [5]	Encoder and Decoder model is used and an additional transcoder is used to convert a sentence into its paraphrasing latent code.	• This model mainly focuses on pattern matching. • Various diverse paraphrased sentences are generated.
arXiv, 2017 [6]	Combination of deep generative models like Variational Autoencoder (VAE) with sequence-to-sequence models (LSTM) to generate paraphrase sentences.	• This model is quite simple and modular. • Sometimes can give out of context paraphrased sentences.

03.REQUIREMENTS

3.1 DESCRIPTION OF REQUIREMENT

To create a tool for paraphrasing of low resource language sentence.

3.2 SOFTWARE REQUIREMENT SPECIFICATION

3.2.1 SCOPE

The **scope** includes the Creation of a Dataset for a low-resource language, Natural language understanding of the sentence, Validating the input sentence, and paraphrasing it along with its Evaluation. The system will **not** include in the scope of the Advanced phases of Natural Language Understanding.

3.2.2 FEATURES

The purpose of the Paraphrasing tool is to provide a paraphrased sentence of low resource language i.e the input sentence will be expressed using different words.
The **key benefit** of the tool is to make the input sentences easier for a user to understand and provide multiple sentences with the same meaning/context as the given input sentence by the user.
The Paraphrasing tool **system description** in brief: The tool will validate if the input sentence meets the requirements and if it does, then paraphrase it and evaluate the same. Also, we can update the dataset whenever necessary.

3.2.3 FUNCTIONAL REQUIREMENTS

There are majorly four functionalities of the paraphrasing tool namely:

- Validate Input: Once the user enters the input sentence, the system will first check if it includes words from the dataset. If it does then the sentence further goes for pre processing phase else the user is prompted to re-enter the sentence.
- Paraphrase Input: Various data cleaning algorithms are performed on this validated sentence like tokenization, stemming, lemmatization, etc. And then by applying the paraphrasing algorithm, multiple paraphrased sentences are generated.
- Evaluate Paraphrased Sentence: The next step after generating paraphrased sentences is to evaluate their similarity index. The lesser the value of similarity index, the closer it is to the original sentence.
- Dataset Updation: The developer can update the dataset with new words or modify the existing words whenever necessary.

1)Validate Input

Use Case	Validate Input
Brief Description	Input is validated against certain conditions as a pre-step in paraphrasing.
Primary Actors	User, Developer
Secondary Actors	None
Preconditions	Input is provided in the required format.
Main Flow:	1. The user provides the input. 2. The input sentence goes through various validation checks. 3. The result is displayed to the user.
Postconditions	The input in validated.
Alternative flows	Prompt to re-enter input.

2)Paraphrase

Use Case	Paraphrase
Brief Description	The provided input sentence is paraphrased as the output.
Primary Actors	User
Secondary Actors	None
Preconditions	Input is validated.
Main Flow:	1. The validated input is sent for preprocessing. 2. After data preprocessing, various paraphrased sentence is generated.. 3. The output is displayed to the user.
Postconditions	Multiple paraphrased sentences are generated..
Alternative flows	None.

3)Evaluate

Use Case	Evaluate
Brief Description	The paraphrased sentence is evaluated under certain metrics.
Primary Actors	Model
Secondary Actors	None
Preconditions	Multiple paraphrased sentences are generated.
Main Flow:	1. The output generated is tested against various metrics . 2. All the generated sentences are accordingly provided with scores.
Postconditions	The most appropriate output can be identified..
Alternative flows	None.

4)Dataset Updation

Use Case	Update Dataset
Brief Description	The dataset is updated with new data or is modified to yield better results.
Primary Actors	Developer
Secondary Actors	None
Preconditions	The dataset is editable.
Main Flow:	1. Check if the proposed change is not present in the dataset. 2. Update the dataset with change.
Postconditions	The model will give better results with the updated data.
Alternative flows	None.

3.2.4 NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements are basically the quality constraints that the system must satisfy. The priority or extent to which these factors are implemented varies from one project to another. They are also called non-behavioural requirements.

3.2.4.1 Performance Requirements

Simultaneous access to the system is expected i.e. the tool should provide the paraphrased sentences whenever users enter the input sentence and simultaneously the developers can also update the dataset.
The time for generating the paraphrased sentence should be less the 60 sec.
The tool should be able to generate a paraphrased sentence that will meet the evaluation criteria.

3.2.4.2 Security Requirements

The database can be accessed only by the developers for updating purposes.

3.2.4.3 Software Quality Attributes

- Accuracy: The tool should always provide accurate results.
- Availability: The tool should be available anytime, anywhere the users require it.
- Portability: The tool should be accessible(ported) on all operating systems and browsers.

- Testability: The tool should provide a matrix to show how closely it matches the input sentence.
- Usability: Any person with minimum typing skills should be able to use the tool.
- Reliability: The tool provides multiple output sentences which will be in context with the input sentence.
- Scalability: The database is scalable i.e. it can be extended any number of times whenever felt necessary.

3.2.5 SYSTEM REQUIREMENTS

3.2.5.1 SOFTWARE REQUIREMENTS

Sr.no	Tool / Technology	Description/ Use
1.	Python	A general-purpose interpreted, interactive, object-oriented, and high-level programming language
2.	nltk	NLTK, or Natural Language Toolkit, is a Python package that you can use for NLP. NLTK supports classification, tokenization, stemming, tagging, parsing, and semantic reasoning functionalities
6.	AST	The ast module (Abstract Syntax Tree) allows us to interact with and modify Python code
7.	Google Colab	Colaboratory, or "Colab" for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education.

Table 3.1 Software Requirement

3.2.5.2 HARDWARE REQUIREMENTS

1. 4 GB RAM
2. Core i3 processor minimum 2.4 GHz frequency (standard)
3. Any operating system

04. SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE

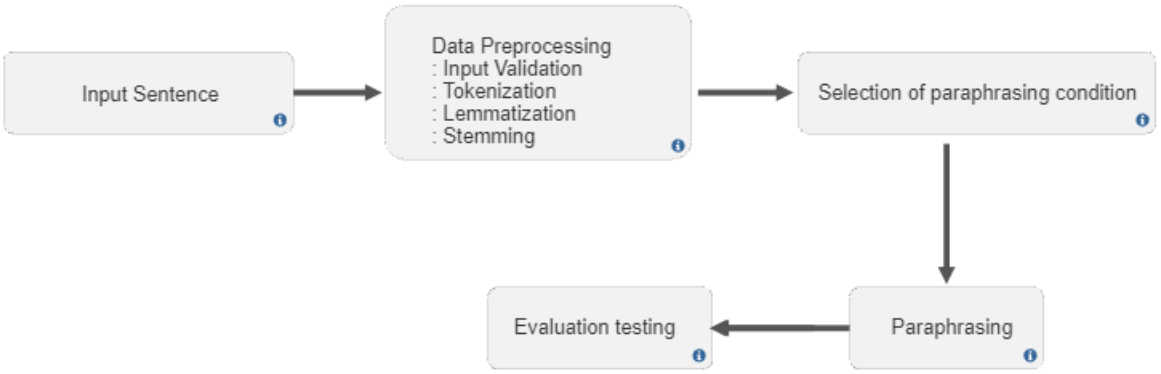


Fig 4.1 - Proposed System Architecture

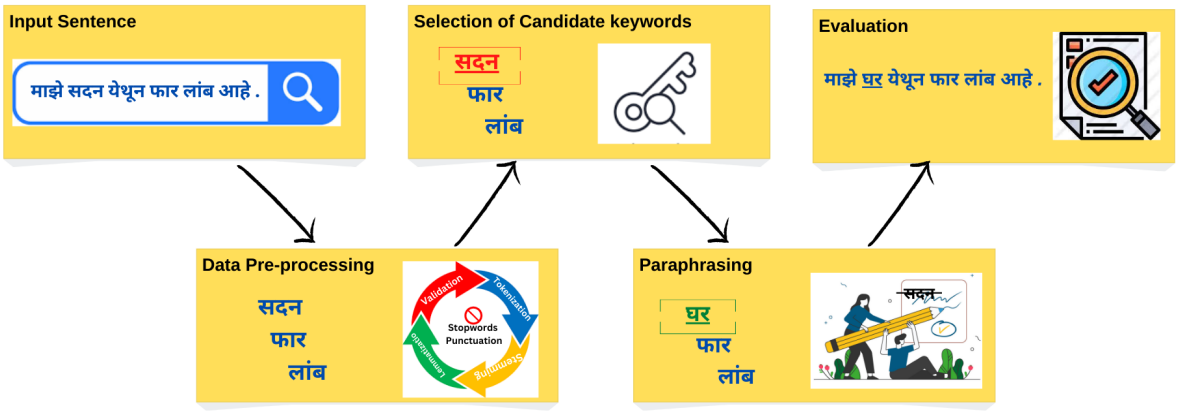


Fig 4.2 - System Flow Diagram

The paraphrasing tool first takes the input sentence from the user and first validates it. The validated sentence then undergoes to the data pre-processing phase which includes tokenization followed by the removing the punctuations then stop word removal followed by stemming and finally lemmatization. After this initial process completes, we look for the keyword which is to be replaced. Various methods can we used for this like TF-IDF or vectorization. Once this process gets completed a list of paraphrased sentences is generated as output along with the similarity index. The lesser the value of similarity index the closer it is to the original sentence.
All the implementation part is performed on Google Collab. The entire code is written in python and various python packages and libraries are used. The nltk package is used for tokenization

and other pre-processing steps. A text file has been used to store data for current use which will be replaced later by a dataset

4.2 DATA FLOW DIAGRAM

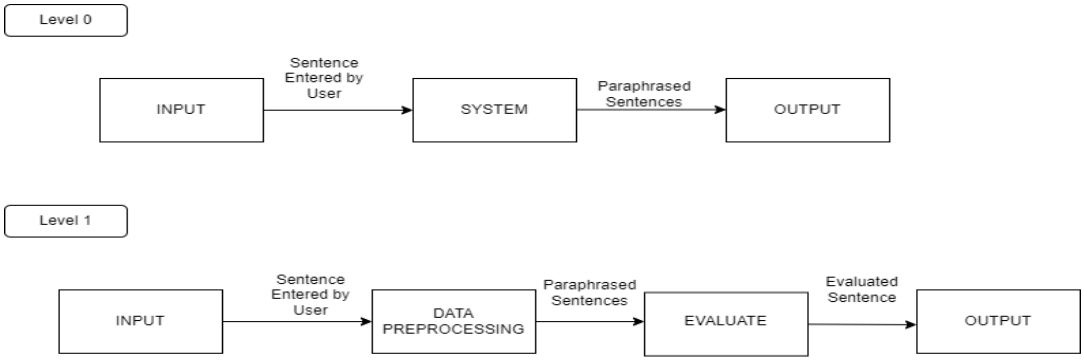


Fig 4.2 - Data flow diagram

4.3 UML DIAGRAM:

4.3.1 USE CASE DIAGRAM:

4.3.1.1. VALIDATE INPUT

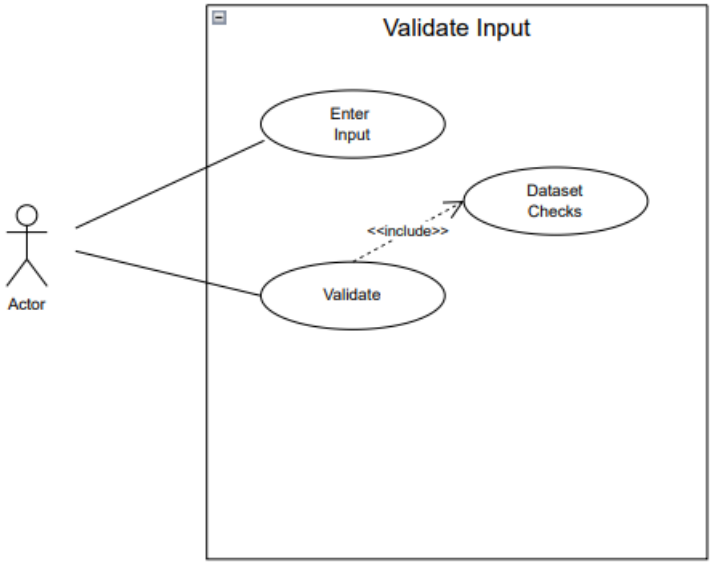


Fig: 4.3.1.1 Validate Input Use Case

4.3.1.2. PARAPHRASING INPUT

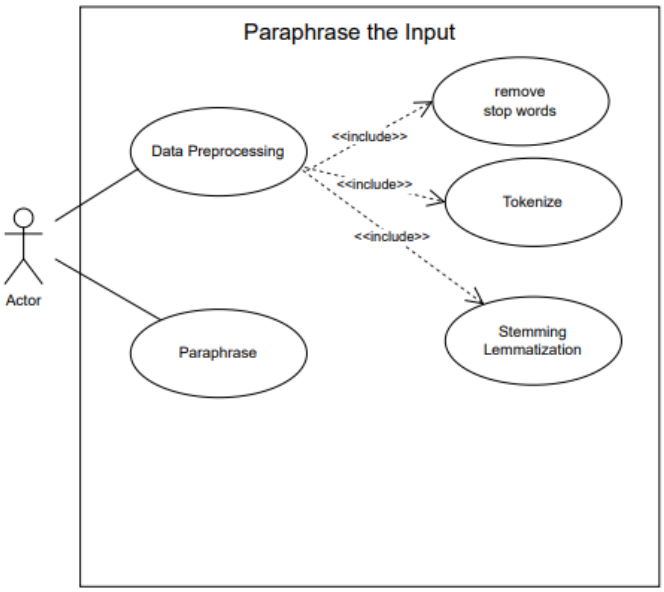


Fig 4.3.1.2 : Paraphrasing Input Use Case

4.3.1.3. EVALUATE OUTPUT

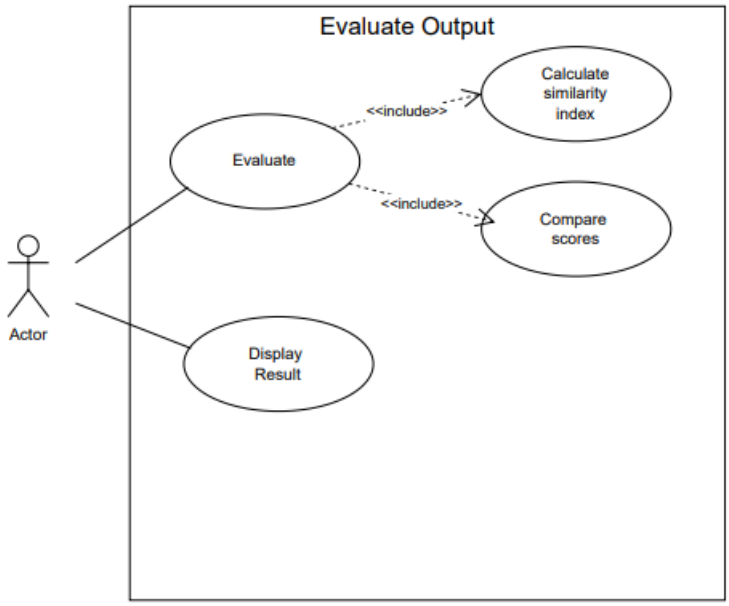


Fig 4.3.1.3: Evaluate Output Use Case

4.3.1.4. DATASET UPDATE

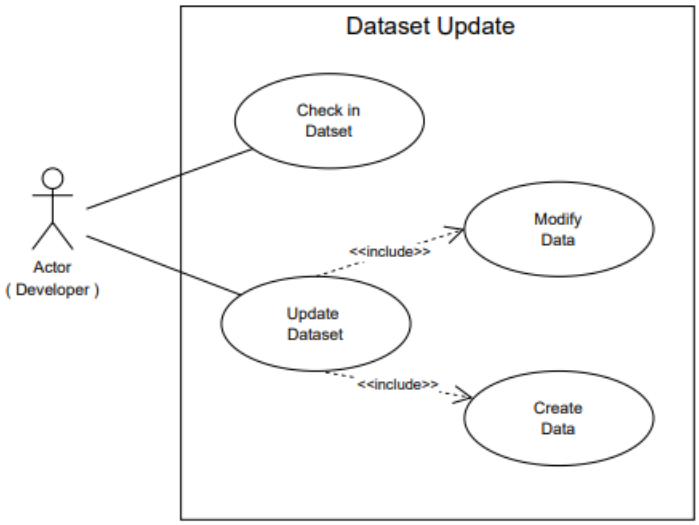


Fig 4.3.1.4. Dataset Update Use Case

4.3.2. STATE DIAGRAM:

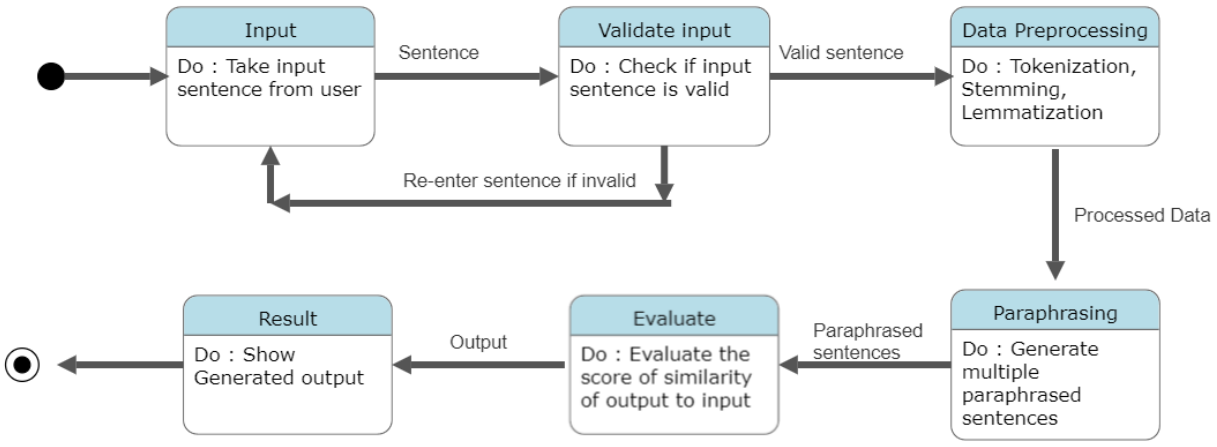


Fig 4.3.2 State Diagram

4.3.3. ACTIVITY DIAGRAM:

Fig 4.3.3 Activity Diagram

4.3.4. SEQUENCE DIAGRAM

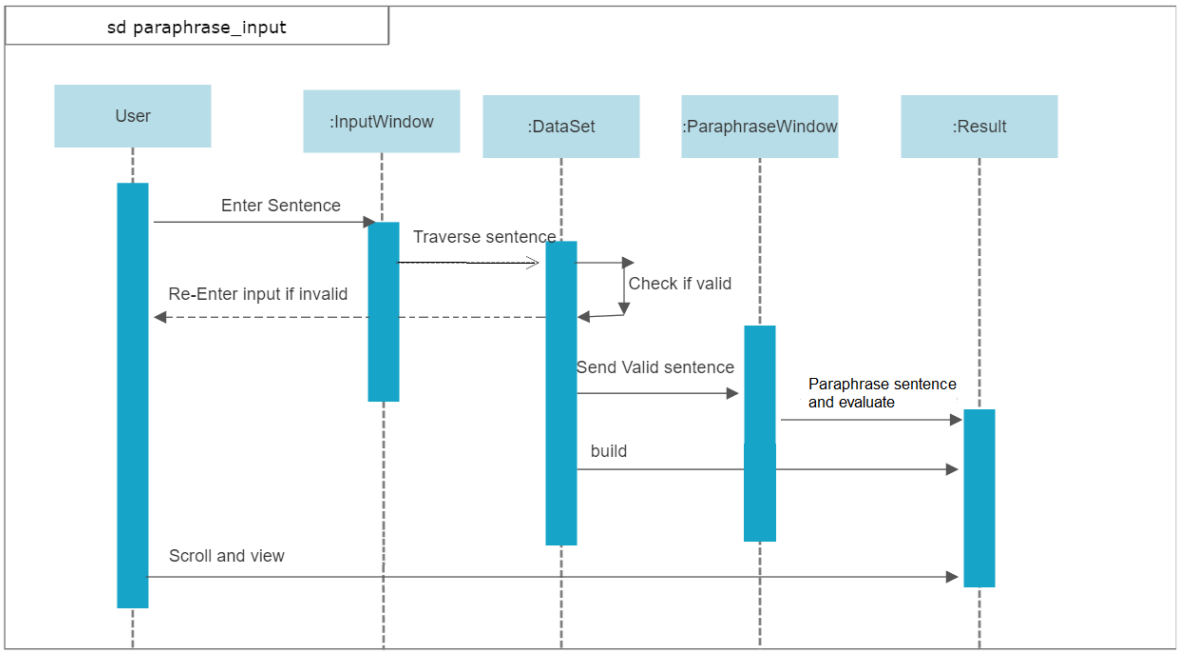


Fig 4.3.4 : Sequence diagram

05 TECHNOLOGY

5.1 TOOLS / TECHNOLOGY

Language : Python
Platform : Google Collab, Spyder
Domain : Natural Language Processing
Tools : nltk, IndoWordNet
Library: mahaNLP
Test : Unit Testing

5.2 TEST PLANS

Precondition/ Assumption	Test case ID	Test case name	Test case description	Test Steps	Expected result	Actual result	Status
Open the application and enter sentence to paraphrase	TC1	Validate input	To check if words entered in sentence are in dataset	1. Enters input in language other than required one 3. Click submit button	A error message " Enter valid input" should be displayed	A success message " Paraphrasing the input" should be displayed	Fail
	TC2	Validate input	To check if words entered in sentence are in dataset	1. Enters input in required language 2. Click submit button	A success message " Paraphrasing the input" should be displayed	A success message " Paraphrasing the input" should be displayed	Pass
Word to add to dataset is not present in dataset	TC3	Update dataset	To update dataset with new word	1. Enters new word to add in dataset 2.Enter submit	A success message "Dataset updated" should be displayed	A success message "Dataset updated" is displayed	Pass
Word to add to dataset is not present in dataset	TC4	Update dataset	To update dataset with new word	1. Enters new word to add in dataset 2.Enter submit	A success message "Dataset updated" should be displayed	None	Fail
Word to add to dataset is present in dataset and new data related to it does not exists	TC5	Modify dataset	To update dataset with word	1. Enters new word with all existing data in dataset, to update in dataset 2.Enter submit	A show success message "Dataset modified"	A success message "Dataset modified" is displayed	Pass

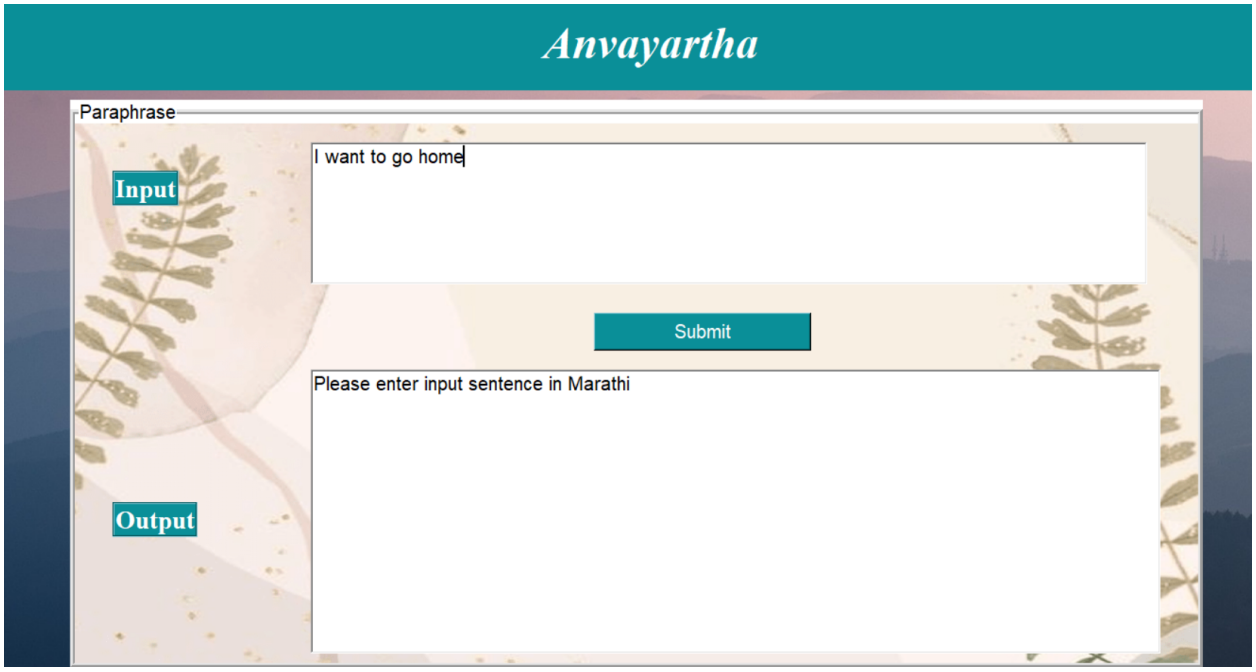
Precondition/ Assumption	Test case ID	Test case name	Test case description	Test Steps	Expected result	Actual result	Status
Word to add to dataset is present in dataset and new data related to it already exists	TC6	Modify dataset	To update dataset with word	1. Enters new word with some new data than existing one in dataset, to update in dataset 2.Enter submit	A message “Data already present in dataset” should be displayed	A success message “Dataset modified” is displayed	Fail
Valid input is given to paraphrase	TC7	Similarity evaluation index	To check if output paraphrased sentence is given input itself	1.Enter Valid Input 2.Click Submit 3.Paraphrased sentence is same to input sentence	Similarity Index of that output paraphrased sentence is 0	Similarity Index more than 0	Fail
Valid input is given to paraphrase	TC4	Similarity evaluation index	To check if output paraphrased sentence is given input itself	1.Enter Valid Input 2.Click Submit 3.Paraphrased sentence is different than input sentence	Similarity Index of that output paraphrased sentence is more than 0	Similarity Index of that output paraphrased sentence is more than 0	Pass

06.IMPLEMENTATION ASPECTS

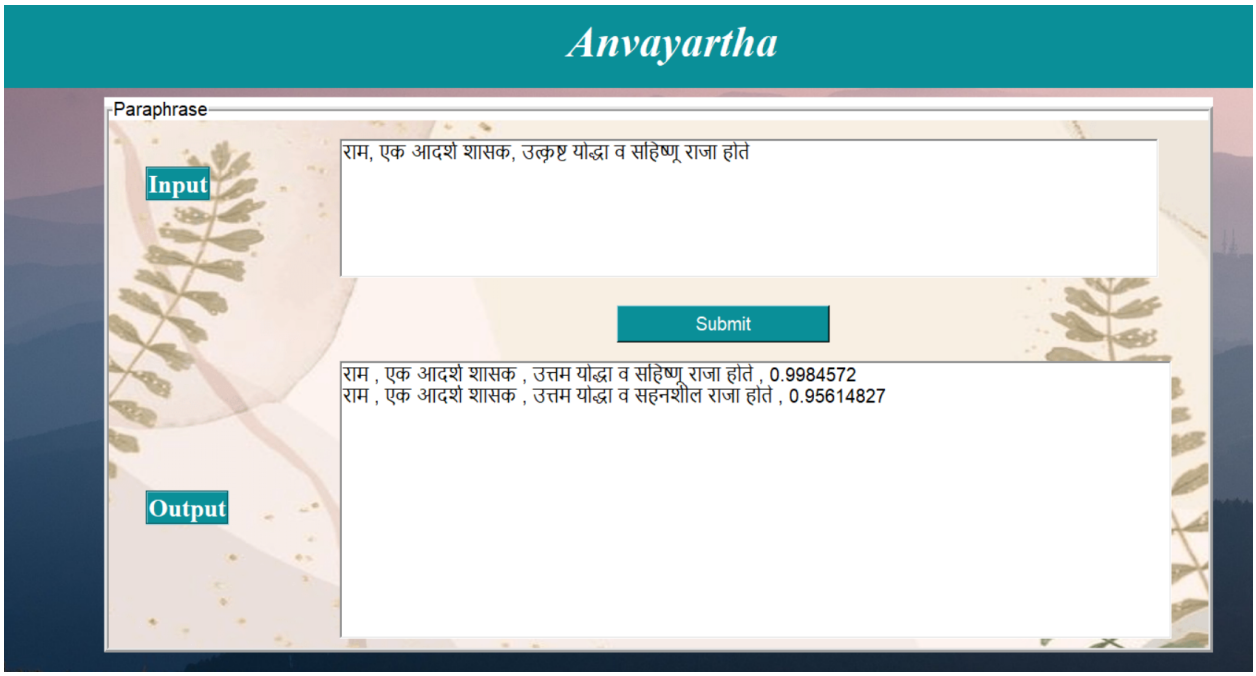
6.1 ALGORITHM

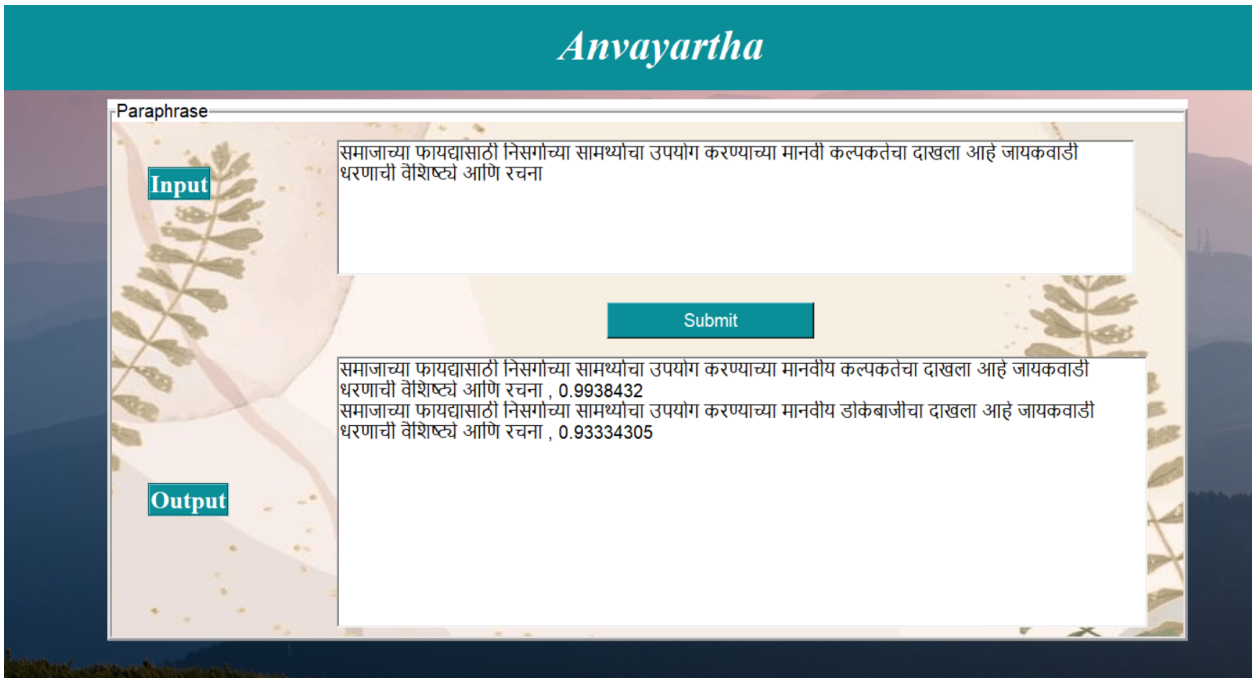
1. Importing libraries
2. Take user input
3. Validate input sentence
   3.1 If sentence is valid
       Go to step 4.
   3.2 else
       Go to step 2.
4. Data pre-processing
   4.1. Tokenization
   4.2 Punctuation Removal
   4.3 Stop word removal
   4.4 Stemming
5. Paraphrasing
6. Evaluation of output
7. Print output

6.2 RESULT


Anvayartha
Paraphrase
Input
I want to go home
Submit
Please enter input sentence in Marathi
Output


Anvayartha
Paraphrase
Input
त्यांच्या वक्तृत्वात गंभीरता आहे
Submit
त्यांच्या वक्तृत्वात स्थिरमनस्कता आहे , 0.7274929
Output


Anvayartha
Paraphrase
Input
राम, एक आदर्श शासक, उत्कृष्ट योद्धा व सहिष्णू राजा होते
Submit
राम , एक आदर्श शासक , उत्तम योद्धा व सहिष्णू राजा होते , 0.9984572
राम , एक आदर्श शासक , उत्तम योद्धा व सहनशील राजा होते , 0.95614827
Output


Anvayartha
Paraphrase
Input
समाजाच्या फायद्यासाठी निसर्गाच्या सामर्थ्याचा उपयोग करण्याच्या मानवी कल्पकतेचा दाखला आहे जायकवाडी धरणाची वैशिष्ट्ये आणि रचना
Submit
समाजाच्या फायद्यासाठी निसर्गाच्या सामर्थ्याचा उपयोग करण्याच्या मानवीय कल्पकतेचा दाखला आहे जायकवाडी धरणाची वैशिष्ट्ये आणि रचना , 0.9938432
समाजाच्या फायद्यासाठी निसर्गाच्या सामर्थ्याचा उपयोग करण्याच्या मानवीय डोकेबाजीचा दाखला आहे जायकवाडी धरणाची वैशिष्ट्ये आणि रचना , 0.93334305
Output

07.CONCLUSION AND FUTURE WORK

7.1 CONCLUSION

Text Generation/ Paraphrasing becomes an important part of NLG because of it's applications in various fields. Also, the low-resource languages still have less internet presence and thus are comparatively complex to be dealt with. Thus, the NLP applications built are unable to extract the context of the sentence leading to partial paraphrasing by word replacement. Every language is different with respect to its grammar, syntax, and linguistic structure. This diversity poses a challenge in utilizing NLP tools for paraphrasing. The paper focuses on various approaches that can be used for the benefit of developing a tool for paraphrasing Marathi language sentences.

The literature survey describes various approaches adopted and used for paraphrasing a sentence(s) by different researchers. The survey shows few kinds of research dealing with low-resource language.

The system/tool can be further enhanced to solve the following issues that need to be addressed: Developing extensive datasets for low-resource languages, and building NLP tools compatible with low-resource languages.

7.2 FUTURE WORK

1. Lemmatization
2. Expansion of dataset

Appendix A: Plagiarism Report of project report

turnitin

Similarity Report ID: oid:8054:36054879

PAPER NAME

G6_Report.pdf

WORD COUNT

2498 Words

CHARACTER COUNT

13506 Characters

PAGE COUNT

28 Pages

FILE SIZE

1.3MB

SUBMISSION DATE

May 24, 2023 11:20 AM GMT+5:30

REPORT DATE

May 24, 2023 11:21 AM GMT+5:30

7% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 6% Internet database
- Crossref database
- 6% Submitted Works database
- 3% Publications database
- Crossref Posted Content database

Excluded from Similarity Report

- Manually excluded text blocks

REFERENCES

[1] Marathi wordnet IIT Bombay:
https://www.cfilt.iitb.ac.in/MobileMarathiWordnet/menglish_version.php

[2] Nandini Sethi, Prateek Agrawal*, Vishu Madaan and Sanjay Kumar Singh.2016. A novel approach to paraphrase hindi sentences using natural language processing

[3] Jianing Zhou, Hongyu Gong, Srihari Nanniyur, and Suma Bhat. 2021b. From solving a problem boldly to cutting the gordian knot: idiomatic text generation. Arxiv preprint arxiv:2104.06541

[4] Lihua Qian, Lin Qiu, Weinan Zhang, Xin Jiang, and Yong Yu. 2019. Exploring diverse expressions for paraphrase generation. In proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp), pages 3164–3173.

[5] Zhecheng An and Sicong Liu. 2019. Towards diverse paraphrase generation using multi-class wassersteingan. Arxiv preprint arxiv:1909.13827.

[6] Ankush Gupta, Arvind Agarwal, Prawaan Singh, and Piyush Rai. 2017. A deep generative framework for paraphrase generation. Arxiv preprint arxiv:1709.05074