International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 06 | June 2019 www.irjet.net p-ISSN: 2395-0072

સારાંશ

Gujarati Text Summarizer

Malvi Shah, Dr. Kalyani Patel

*[*1]Department of Computer Science, Rollwala Computer Center, Gujarat University, Ahmedabad, Gujarat, India*
[2]K.S school of business management, Gujarat University, Ahmedabad, Gujarat, India

---

Abstract - Gujarati Text Summarizer is a task to generate a succinct and lucid summary of a Gujarati text fetching its key information content and overall meaning. The Summary produced by system allows readers to quickly and easily understand what the text is all about. The overall intension of developing a text summarizer is that readers can choose a specific content of their need from the abundance of material available in Gujarati. Gujarati text summarizer gives a short summary of the Gujarati text which makes easy for reader to choose material of their need without wasting time is reading all text.

Gujarati text summarization is very challenging, because when we as humans summarize a piece of text, we usually read it entirely to develop our understanding on the basis of its sentence formation, words, noun, Gender and Grammar , and then write a summary highlighting its main points. Since computers lack human knowledge and language capability especially on Indian Language like Gujarati, it makes Gujarat text summarization a very difficult and non-trivial task.

Keywords - Gujarati text summarization, Text summarization, Automatic summarization., NLTK, Tokenize, Gujarati Stopwords, Ubuntu.

Introduction

Automatic summarization is a reliable and effective way to resolve the information overload problem. Automatic summarization is a hot topic of research nowadays, only very few software tools are available to the end users and none of them are particularly designed for Gujarat language.

Text summarization is technique of compressing a text into a summary. The brief summary produced by summarization system allows readers to quickly and easily understand the content of original text without having to read each whole text. The overall motive of text summarization is to convey the meaning of text by using less number of words and sentences

Gujarati text Summarization is a useful tool for selecting relevant text, and for extracting the key points of text. The goal of automatic Gujarati text summarization is to take an information source, extract content from it, and present the most important content in a condensed form. There is an abundance of Gujarati material available, usually information is more than needed. Therefore, a twofold problem is encountered: (1)Searching for relevant text through an overwhelming number of text available, and (2) absorbing a large quantity of relevant information.

For Gujarati language, no full-fledged text summarizer exists yet. Also due to resource constraint and its characteristics, existing summarizers cannot be adopted for Gujarati language. a swarm of information comes from the Internet in a Gujarati textual form as well. Some of the distinctions, with respect to computational linguistics, between Gujarati and English language were found which are specified below: Gujarati follows SOV as its default sentence structure as It has free word order.

Gujarati text summarization includes one phase for sentiment analysis which generates emotion related to the text.. Sentiment includes Positive, Negative and Neutral based on polarity of text.

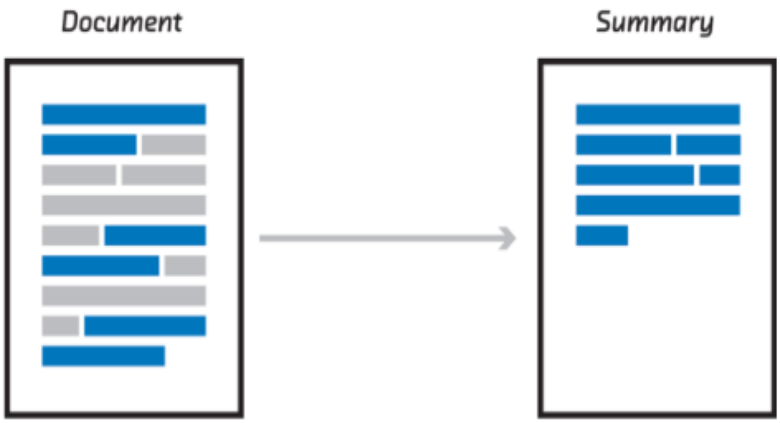
There are basically two types of summarization:

→ **Abstractive** : The selected document sentences are combined coherently and compressed to exclude unimportant sections of the sentences.

→**Extractive**: Extracts are produced by identifying important sentences which are directly selected from the document.

1.1 What is Extractive type?

Extractive text summarization techniques perform summarization by picking portions of texts and constructing a summary, unlike abstractive techniques which conceptualize a summary and paraphrases it.



Extractive summarizers

Languages and Libraries

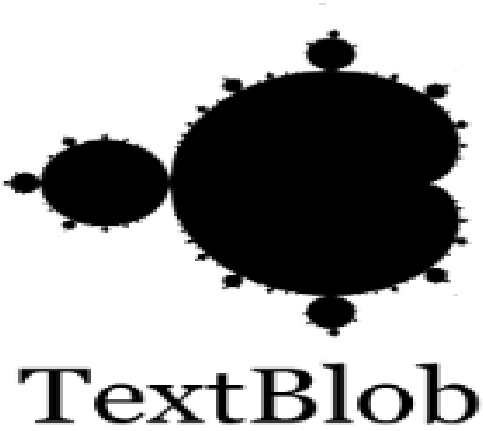
2.1 Python

Python is an interpreted, high-level, general-purpose programming language. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

2.2 Libraries:

2.2.1 Textblob

TextBlob is a Python library for processing textual data and for sentiment analysis. The scale of the words' polarity consisted of three degrees: +1 for positive words, and -1 for negatives words. Neutral words will have a score of 0.



2.2.2 Gensim

Gensim is a free Python library designed to automatically extract semantic topics from documents, as efficiently (computer-wise) and painlessly (human-wise) as possible.

Text Processing

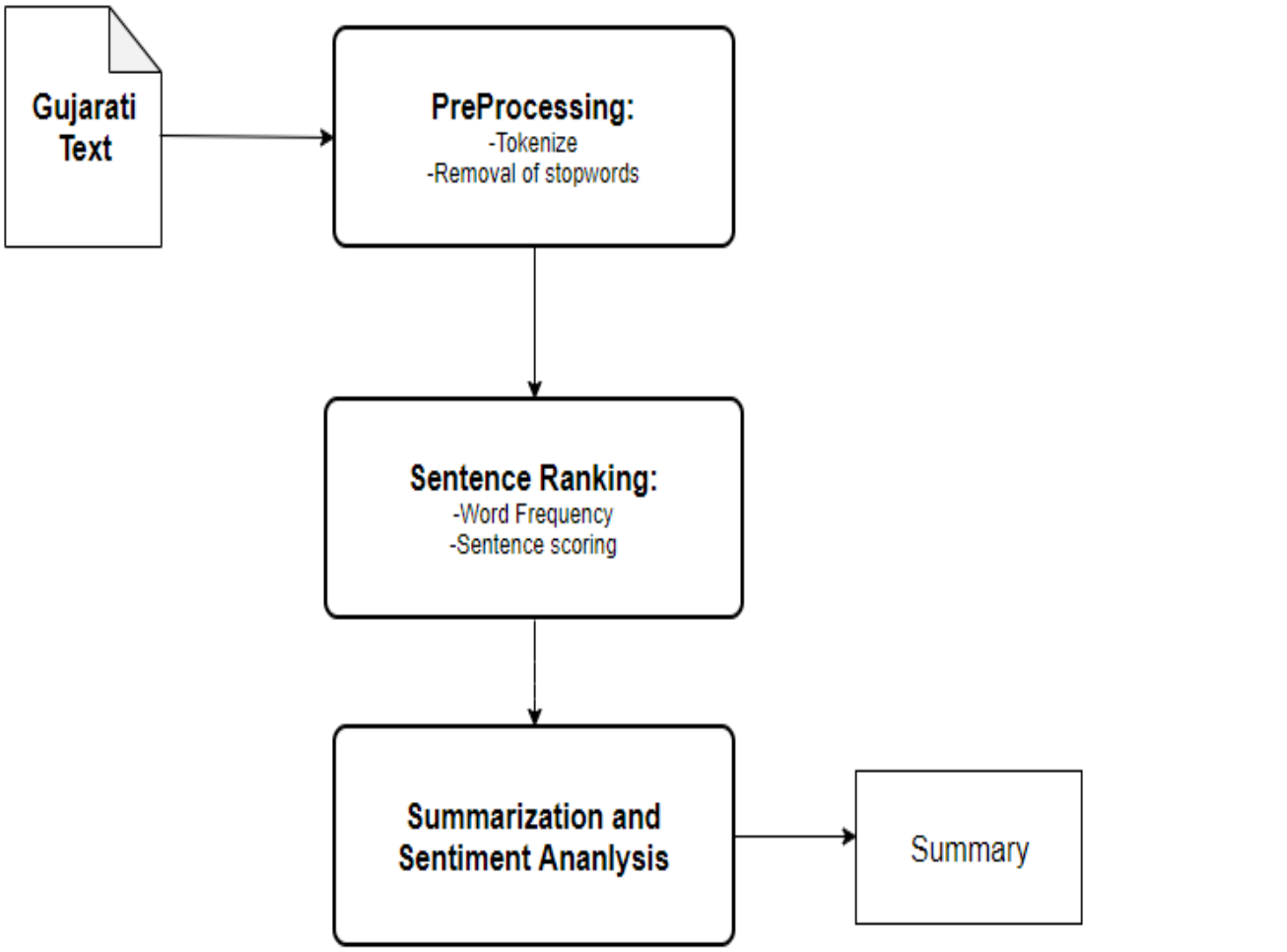
3.1 The summarization process can be decomposed into three phases:

1-Preprocessing.

2-Sentence ranking.

3-Summary generation.

Block Diagram:



3.1.1 Pre Processing By NLTK:

- **NLTK** is a leading platform for building Python programs to work with human language data.
- **Tokenization of words**:Tokenization is the process by which big quantity of text is divided into smaller parts called tokens.
- **Stop Words**: A stop word is a commonly used word (such as “the”, “a”, “an”, “in”) that a search engine has been programmed to ignore, both when indexing entries for searching and when retrieving them as the result of a search query.
- **POS**-The POS tagger in the NLTK library outputs specific tags for certain words. The list of POS tags is as follows, with examples of what each POS stands for.

3.1.2 Sentence Scoring:

- The Summarizer with arrange the sentences on the basis of its feature and the weightage of the sentence.

3.1.3 Summarization and Sentiment Analysis:

- TextBlob aims to provide access to common text-processing operations through a familiar interface.

- Gensim is designed to process raw, unstructured digital texts

*Installing Python 3,O.S-Ubuntu.

1.Install NLTK.

2.import NLTK.

3.import function for tokenization of words and sentence tokenization.

4. import stopwords, for Gujarati we have to make own text file including all stopwords .

5.pip install textblob.

6.pip install gensim.

7.make .py file importing all libraries (textblob, gensim)

8.write commands in python for input data, tokenization, removing stopwords, commands of Textblob(sentiment analysis) and gensin summarization (summarizing text).

Challenges

- Nearly 50 million people of western part of India use Gujarati language. About 65.5 million speakers of Gujarati exists worldwide, making it the 26th most spoken native language in the world. Before the model for text summarization of Gujarati language was developed, a strong study related to why English text summarization system cannot used for Gujarati language was done where this difference was seen.
- Some of the distinctions, with respect to computational linguistics, between Gujarati and English language were found which are specified below: Gujarati follows SOV as its default sentence structure. It has free word order i.e. words can move freely within a sentence without changing its meaning. For example, for an event described as “Tina learns from Sneha” can be written in more than one ways in Gujarati as follows: ટીના સ્નેહા પાસેથી શીખે છે , સ્નેહા ટીનાને શીખવે છે.
- It has relatively rich set of morphological variants. A word may appear with a number of inflections and each inflection can appear with several words. કરવું, , કર્યું , કરે છે , કરો , તુ કર , ફરવુ etc.
- Verbs undergo morphological changes depending upon the number and gender.

(1) રામ સૂઈ રહ્યો છે.

(2) રીના ઊંઘી રહી છે.

(3) માતાપિતા સૂઈ રહ્યા છે.

- Adjectives may appear with variations to agree with gender.
- The complex predicates change the functional structure of the sentence.

Evaluation

- **QUALITY EVALUATION:**

One way to assess the quality of summary is to ask Human Judges to grade for its READABILITY and ACCEPTABILITY.

- **INFORMATIVENESS:**

How informative a summary is ,it is measured in term of the amount of information preserved from the source text at different level of comparison or from ideal summary.

- **Sentence Recall:**

It measures the Fraction of sentences in the ideal summary that have been recalled in the Automatically Generated Summary**.**

Conclusion

In this paper, a method to summarize Gujarati text is presented. The performance of text summarization improves by adding linguistics components to it. Though human generated summaries are difficult, still using linguistic components like Stemmer and String similarity measure, summary with good recall can be achieved. The summary can be enhanced by using approaches of Machine learning and different NLP tools.

References

1. Pre-Processing Phase of Text Summarization Based on Gujarati Language.International Journal of Innovative Research in Computer Science & Technology (IJIRCST) ISSN: 2347-5552, Volume-2, Issue-4, July – 2014.

2. Saaraansh: Gujarati Text Summarization System.IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: 2249-9555 Vol.7, No.3, May-June 2017.

3. String Similarity Based on Phonetic in GUJARATI LANGUAGE using Gujsim AlgorithmInternational Journal of Computer Engineering and Applications, Volume XII, Issue III, March 18, www.ijcea.com ISSN 2321-3469.

4. A Review on Automatic Text Summarization Based on Gujarati Language.NCI2TM: 2015, 978-81-927230-9-9.

5. A REVIEW PAPER ON TEXT SUMMARIZATION OF HINDI DOCUMENTS.INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS ISSN 2320-7345 Vol.3 Issue.5, Pg.: 89-92 May 2015 .

6. Extractive Based Automatic Text Summarization.Journal of Computers Volume 12, Number 6, November 2017.

7. https://www.nltk.org/

8. https://textblob.readthedocs.io/en/dev/

Acknowledgments

Dr. Savita Gandhi [H.O.D – Department of Computer Science]

Dr.Jyoti Pareek [Department of Computer Science]

Shri. Hiren Joshi [Department of Computer Science]

Shri. Hardik Joshi [Department of Computer Science]