INTERIM PROJECT REPORT

October, 2019

Project Scope

The focal aim of the project is to create an ***extractive text summarizer***. Hence, the goal is to process a given article and isolate a subset of sentences from the text itself which cover the overall gist of the document.
Being primarily an **experimental** project, the main focus will be on exploring methods to generate better summaries, more than creating a finished application.

Deliverables

The following are aimed to be presented in their entirety by the final project deadline, therefore some of the given below entities may be absent from the interim submission or may not be in their final form yet:

- Working code (.py)
- Project report
- Detailed report on relevant research papers
- Readme for code

Up Till Now

Research Papers

We have heretofore selected two research papers pertaining to the project:

- Sentence Extraction Based Single Document Summarization [link]
- Rule Based Summarization for Marathi Text [link]

Short explanations regarding both have been provided later in the document. The final report will contain a more **detailed report on all the research papers**. Till now, we have selected only two papers, but we will be exploring for more sources (possibly even replace some of the existing papers) by the final project deadline.

Code

- We have prepared a **preliminary (working) code** based primarily on the first paper. It focuses on **frequency distribution of words** and **positions of sentences** in the given text as focal features. A more detailed explanation will be provided in the final project report.
- The code (for the time being) accepts thoroughly cleaned and unsegmented text as an input. (sample inputs have been provided)
- Information regarding the usage of the code has been provided in a README

Upcoming Agenda

Paradigm

- The approach is presently very rudimentary and purely analytical. We plan to incorporate more dynamic and useful **features** (both **mathematical and linguistic**) for sentence ranking and overall selection (possibly sentential similarity, sentence structure & semantic relations)
- In the upcoming submission, we also plan to have a **dataset of article-summary pairs** to compare the output of our programme and attempt to calculate the **accuracy** of our approach

Code

- Currently accepts only cleaned input. Plan to make the code more robust in terms of both **preprocessing** and also **extracting more information** regarding the document from document formatting (e.g. bullet points, headings, bold/italicized text)

Research Papers

- As mentioned previously, we plan to **incorporate more linguistic features** in sentence ranking, for which we will be exploring **more research sources** and hence including more papers and consequently, detailed reports, in our final project submission

Short Paper Descriptions

Sentence Extraction Based Single Document Summarization

The following **pipeline** gives a rough idea of the paper's approach towards summarization

1. Basic preprocessing, tokenization and sentence marking
2. NE recognition and POS tagging
3. Feature extraction
    a. Sentence Level Features
        i. Sentence position in document
        ii. Referring pronouns
        iii. Length of sentences
    b. Word Level Features
        i. Term frequency
        ii. Word length
        iii. POS tag
        iv. NE tag
        v. Word Familiarity
4. Sentence ranking (on the basis of above features)
5. Sentence selection (on the basis of-)
    a. Coherence factor
    b. Discourse connectors
6. Final summary refinement

We are trying to adhere to a similar pipeline as above, with many additions and deletions of course.

Rule Based Summarization for Marathi Text

The paper follows a slightly different and interesting approach towards text summarization by using question extraction.

- It first uses a rule based model to extract questions from the input text, then ranks them in order of relevance and finally adds answers to the highest ranked *x* questions as the summary.

Brief Overview:

- Input Text -> Tokenizer -> POS Tag (includes NER) -> Stemming -> Rule Based Question Generation Framework -> Question Ranking -> Answer Extraction and Summary
- The question generation framework uses basic syntactic rules of the language to form questions of the types <who, where, how much/many, when, what>.
- These questions were then ranked based on frequency of generation, type and other parameters. Noun-based questions were given a higher priority as it yielded to better results when summarizing by adding the answers to the highest ranked questions.

Contributions

Monil Gokani (2018114001) - Exploring and evaluating research papers based on different Indian Languages and looking for ways to adapt them to Hindi, Looking for ways to incorporate linguistic features into “pure frequency based models” find better word.

KV Aditya Srivatsa (2018114018) - Studying existing summarization algorithms and implementing them in actual code. Also, studying research papers to come up with new approaches and using NL modules and toolkits.