**Template for CIS-530 Final Project**

|  |  |  |
| --- | --- | --- |
| **Shruthi Gorantala** | **Karan Pradhan** | **Harshitha Yenugula** |
| [shruthig@seas.upenn.edu](mailto:shruthig@seas.upenn.edu) | [karanpr@seas.upenn.edu](mailto:karanpr@seas.upenn.edu) | [yenugula@seas.upenn.edu](mailto:yenugula@seas.upenn.edu) |

|  |  |
| --- | --- |
|  |  |

Abstract

This document presents the implementation details of the three summarizers - TF\*IDF System, LexPageRank, KL divergence system, along with their rouge scores. A thorough analysis and inferences drawn while implementing the summarizers mentioned is also depicted.

1 At the Beginning

* 1. Fonts

If you are confused about what font size you should use, just follow this template, using 11pts. Put footnotes at the bottom of the page. They may be numbered or referred to by asterisks or other symbols.[[1]](#footnote-1) Footnotes should be separated from the text by a line,[[2]](#footnote-2) with 9 point font.

* 1. Graphics

**Illustrations:** Place figures, tables, and photographs in the paper near where they are first discussed, rather than at the end, if possible. Wide illustrations may run across both columns.

1. Basic Systems

Illustrate your approaches and choice of parameters for the three Basic Systems. Apart from state what you've decided to use, also explain why you made that choice.

* 1. TF\*IDF System
  2. LexPageRank System

LexPageRank is based on eigen vector centrality. In this model, we construct a graph based on cosine similarity. The nodes in the graph are the individual sentences and they are connected to other nodes( sentences) if the cosine similarity between them is above a certain threshold. The centrality of a node is called the Page Rank in this system. It is calculated using the recursive formula

*PR(A) = (1-d)/N + d (PR(T1)/C(T1) + ... + PR(Tn)/C(Tn))*

Where *PR(A)* is the page rank of *A* and *T1..Tn* are the nodes connected to *A. N* is the number of nodes. *C(Ti)* gives the number of outgoing edges from page *Ti* . However in our case the graph is undirected thus *C(Ti)* gives the degree of the node *Ti.* The damping factor is *d*. We have used the damping factor as 0.85 from a few observations. We have converted this recursive formula to a iterative one and the iterations terminates when we have a minimum k-error ( converging condition ). The threshold for construction of the graph is chosen as 0.3 so that the graph is dense enough. The k-error is chosen as 0.001 (squared error). If the algorithm does not converge then we iterate 75 times.

* 1. KL Divergence System

KL divergence is calculated between input unigrams and the unigrams of the summarizer built so far. This formula is used for selecting the next sentence for the summary with minimum divergence from the set of summary sentences already at hand. The denominator Q(w) ensures that the sentences that have frequently occurring words w.r.t to the entire tokens taken form multiple documents are not considered for selection into the summarizer (as the ln (P(w)/Q(w)) value becomes negative when Q(w) is higher than P(w)). While implementing the algorithm we ensured to take the unique set of input sentences and also marking off the sentences already added selected( to avoid adding the same sentence multiple times)

* 1. Performance on Development Set

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| System | TF\*IDF | LexRank | KL Divergence | K-summarizer |
| Rouge-2 Recall | 0.07901 | 0.073 | 0.06326 | 0.0668 |

1. Your Summarization System

Discuss how you implement your summarization system in this section.

* 1. System Design

This part shows general idea of your system. You may use flowchart, graphics or pseudo-code to describe your algorithm.

* 1. Resources & Tools Used

What resources or tools you have used and how they are included in your implementations.

Example: Wordnet, Stanford-Parser, MPQA.

I use Stanford-Parser in order to help…

* 1. Performance

1. Discussion and Analysis

References

Alfred. V. Aho and Jeffrey D. Ullman. 1972. *The Theory of Parsing, Translation and Compiling*, volume 1. Prentice-Hall, Englewood Cliffs, NJ.

American Psychological Association. 1983. *Publications Manual.* American Psychological Association, Washington, DC.

Association for Computing Machinery. 1983. *Computing Reviews*, 24(11):503-512.

Ashok K. Chandra, Dexter C. Kozen, and Larry J.Stockmeyer. 1981. Alternation. *Journal of the Association for Computing Machinery*, 28(1):114-133.

Dan Gusfield. 1997. *Algorithms on Strings, Trees and Sequences*. Cambridge University Press, Cambridge, UK.

1. This is how a footnote should appear. [↑](#footnote-ref-1)
2. Note the line separating the footnotes from the text. [↑](#footnote-ref-2)