NATURAL LANGUAGE PROCESSING
CSC 4309, Section 01
Topics: Project Presentation

Lecturer: Dr. SURIANI BT. SULAIMAN

Presented by :

Name: Hasan Tanveer Mahmood

Matric: 1725413

Name: Jaki Fayek Alvi Rahman

Matric: 1721485

Name: Md Raisul Islam

Matric: 1725501

Name: Liu Yufei

Matric: 1722279

Introduction :

Title: Text Summarization Using Tf-Idf Model

- The goal of summarization is to capture the important information contained in large volumes of text, and present it in a brief, representative, and consistent summary.
- In this project we introduce an application of Tf-Idf model to the problem of multi-document summarization.

Introduction :

- Extractive summarization works as follows:

  Input document -> Finding most important words from the document -> Finding sentence scores based on important words ->Choosing the most important sentences based on scores obtained.

- Experiments will show that improve the performance of a state-of-the-art summarization framework which strongly indicate the benefits of TF-IDF model for this task.

Related Work :

1. Extractive based Text Summarization Using K-Means and TF-IDF BY Rahim Khan, Yurong Qian, Sajid Naeem in 2019.

- Method : K-means and TF-IDF (Term Frequency-Inverse Document Frequency).

  Used the K-means and TF-IDF method for extractive text summarization with K value predefined.

Related Work :

2. Automatic Text Summarization Using Gensim Word2Vec and K-Means Clustering Algorithm
BY Mofiz Mojib Haider, Md. Arman Hossin etc. in 2020

- Method : Word2Vec and K-Means Clustering algorithm

intended to automatically extract semantic topics from documents in the most efficient way possible

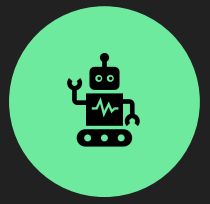
Related Work :

3. Extractive summarization using continuous vector space models BY Mikael K°ageb¨ack, Olof Mogren, Nina Tahmasebi, Devdatt Dubhashi in 2014

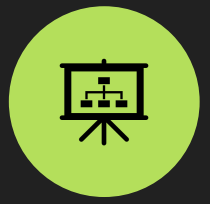
- Method : vector space models.

This model important for improve the performance of a state-of-the-art summarization framework .

Technical Background:



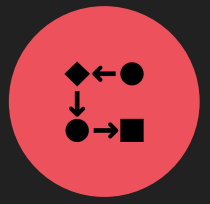
NLTK



REUTERS CORPUS

Tf- Idf

POSITION OF SENTENCE



EVALUATION

NLTK
(Natural Language Toolkit)

A leading platform for building Python programs to work with ***human language*** data.

Text Processing

- Document Categorization/Topic Detection
- Phrase Extraction/Summarization
- Frequency Analysis
- Sentence & Word Tokenization
- Part-of-speech Tagging
- Text Classification
- (etc.)

NLTK is a suite of libraries and programs for symbolic and statistical NLP for English written in the Python programming language.

NLTK supports classification, tokenization, stemming, tagging, parsing, and semantic reasoning functionalities

Over 50 corpora and lexical resources such as WordNet.

REUTERS CORPUS

What is available?

1. **RCV1**
2. **RCV2**
3. **TRC2**

- **RCV1**

Reuters Corpus, Volume 1, English language, 1996-08-20 to 1997-08-19 (Release date 2000-11-03, Format version 1, correction level 0)

- **RCV2**

Reuters Corpus, Volume 2, Multilingual Corpus, 1996-08-20 to 1997-08-19 (Release date 2005-05-31, Format version 1, correction level 0)

- **TRC2**

Thomson Reuters Text Research Collection (TRC2), 2008-01-01 00:00:03 to 2009-02-28 23:54:14

Tf- Idf

- tf (term frequency)
- idf (inverse document frequency)

tf(t)

= Number of times term t appears in a document / Total number of terms in the document

idf(t)

= $\log_e$ (Total number of documents / Number of documents with term t in it)

4 features:

1. Headline
2. Length
3. Position
4. tf-idf

POSITION OF SENTENCE

Values borrowed from:

https://github.com/xiaoxu193/PyTeaser

The position of a sentence in a document has been traditionally considered an indicator of the relevance of the sentence, and therefore it is frequently used by automatic summarization systems as an attribute for sentence selection. Sentences close to the beginning of the document are supposed to deal with the main topic and thus are selected for the summary. This criterion has shown to be very effective when summarizing some types of documents, such as news items.

EVALUATION

Bilingual Evaluation Understudy, is a score for comparing a candidate translation of text to one or more reference translations.

“The closer a machine translation is to a professional human translation, the better it is".

BLEU-4

- 1-gram(25%)
- 2-gram(25%)
- 3-gram(25%)
- 4-gram(25%)

Length

- Length - 3
- Length - 5
- Length - 8

ROUGE

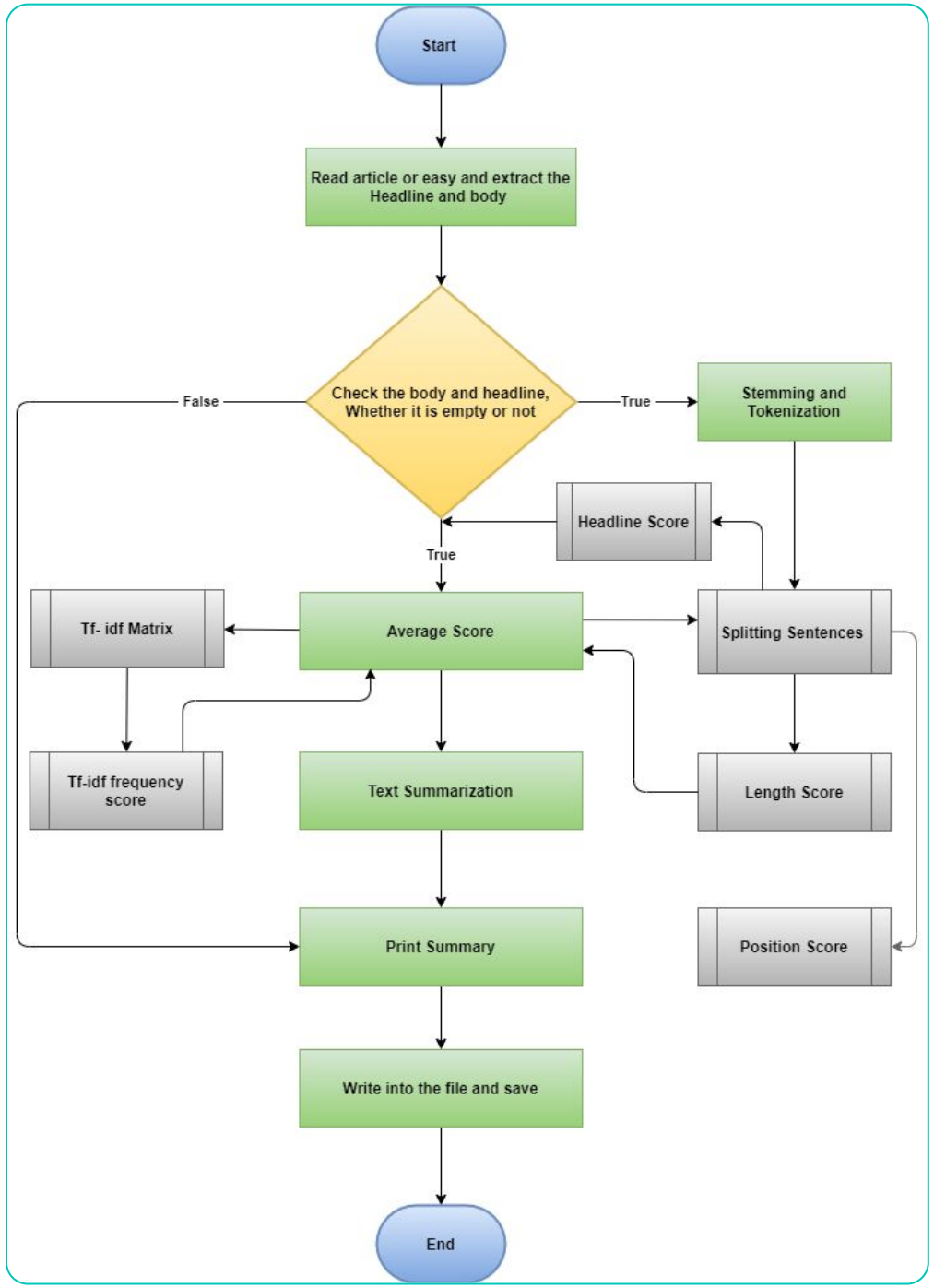
- ROUGE-1:

  the overlap of unigram (each word) between the system and reference summaries.
- ROUGE-2:

  the overlap of bigrams between the system and reference summaries.
- ROUGE-L:

  Longest Common Subsequence (LCS) based statistics.

Methodology:



Model Creation :

```
class Summarizer():

    #taking in two articles and extracting the headline and body to place them in a list
    def __init__(self, article):
      self._articles = []
      for doc in article:
            with open(doc) as f:
                headline = f.readline() #first line of article is the headline
                url = f.readline() #second line of article is the url
                body = f.read().replace('\n', ' ') #read the remaining of the article and replace the empty lines with a whitespace
                #if headline and body is not empty, then we assign the values into 'articles' list
                if not self.valid_input(headline, body):
                    self._articles.append((None, None))
                    continue
                self._articles.append((headline, body))


    #check if headline and body has any text or not
    def valid_input(self, headline, article_text):
        return headline != '' and article_text != ''
```

```
Individual n-grams for length-3.
---------------------------------
1-gram: 0.035714
2-gram: 1.000000
3-gram: 1.000000
4-gram: 1.000000
BLEU Scores:  0.4347208719449914
```

Individual n-gram of Summary length-3

```
Individual n-grams for length-5.
---------------------------------
1-gram: 0.036607
2-gram: 1.000000
3-gram: 1.000000
4-gram: 1.000000
BLEU Scores:  0.43741277066697004
```

Individual n-gram of Summary length-5

```
Individual n-grams for length-8.
---------------------------------
1-gram: 0.044643
2-gram: 1.000000
3-gram: 1.000000
4-gram: 1.000000
BLEU Scores:  0.45966135761245924
```

Individual n-gram of Summary length-8

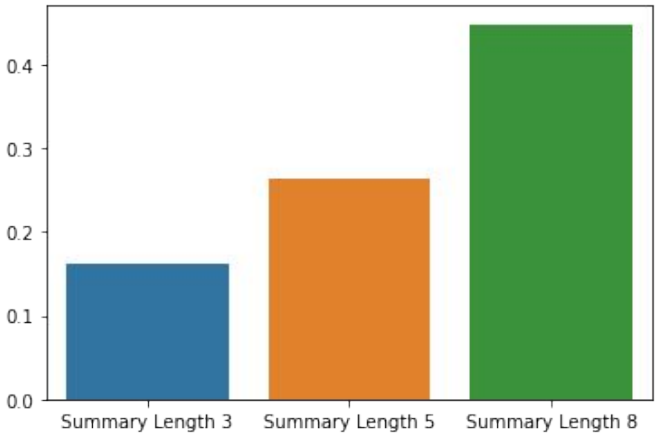
Analysis & Result:

```
Rouge Scores for length-3

rouge-1
----------------------
f 0.16101694544527442
p 0.10674157303370786
r 0.3275862068965517

rouge-2
----------------------
f 0.03418803050295898
p 0.022598870056497175
r 0.07017543859649122

rouge-l
----------------------
f 0.15789473277340732
p 0.11029411764705882
r 0.2777777777777778
```

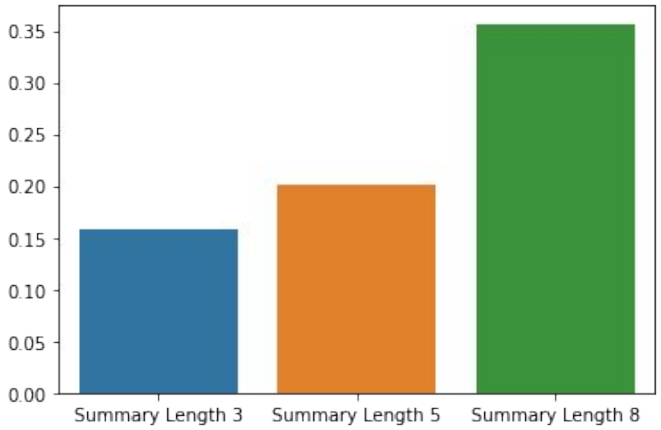
```
Rouge Scores for length-5

rouge-1
----------------------
f 0.2631578903092318
p 0.19662921348314608
r 0.3977272727272727

rouge-2
----------------------
f 0.05303029861139844
p 0.03954802259887006
r 0.08045977011494253

rouge-l
----------------------
f 0.20095693325335967
p 0.15441176470588236
r 0.2876712328767123
```

```
Rouge Scores for length-8

rouge-1
----------------------
f 0.4475920629890297
p 0.4438202247191011
r 0.4514285714285714

rouge-2
----------------------
f 0.20512820012857053
p 0.2033898305084746
r 0.20689655172413793

rouge-l
----------------------
f 0.3565891423015444
p 0.3382352941176471
r 0.3770491803278688
```

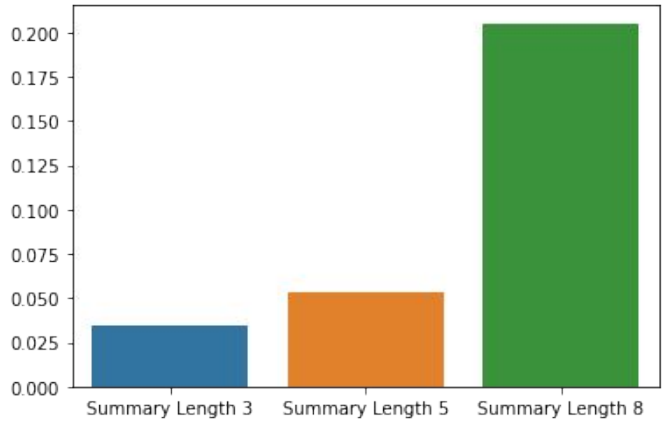
Analysis & Result (cont.):



F-1 Scores of Rogue-1



F-1 Scores of Rogue-2

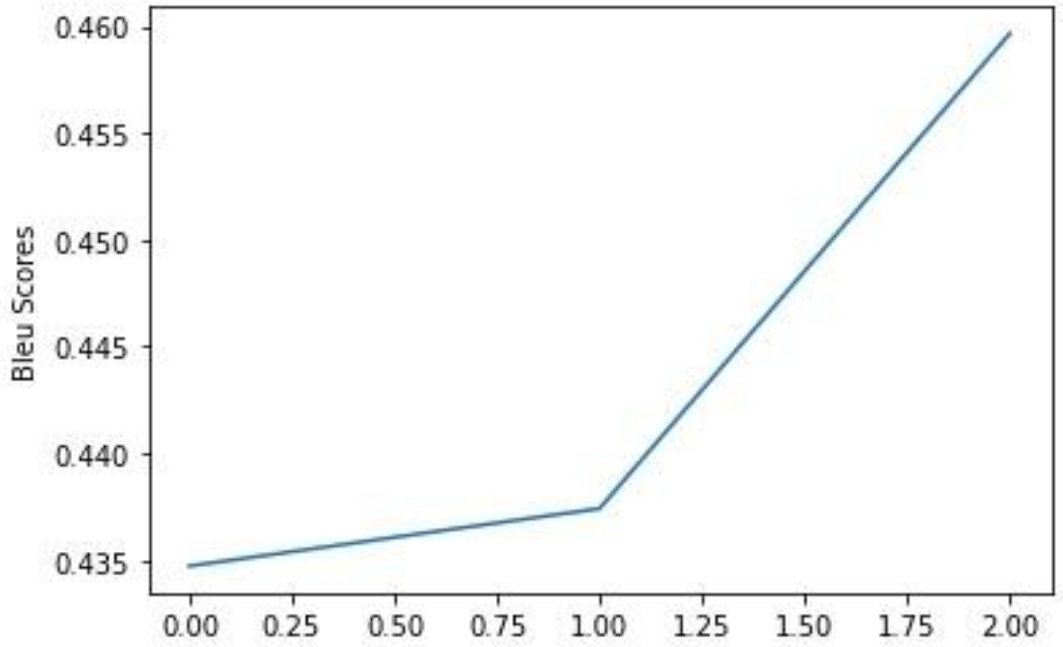


F-1 Scores of Rogue- l

Analysis & Result (cont.):

Analysis & Result (cont.):

Bleu Scores of Individual lengths (3,5 & 8)



Conclusion

According to the above analysis of BLEU Scores and Rouge Scores we can conclude that our model works best for summary length 8. In the Rouge scores summary length 8 is also ahead of summary length 3 and 5. Thus, we can consider our model is best for 8 lines summary. In the future, we will try to implement machine learning algorithms in our model to upgrade our model as well as performance.

References:

- https://www.malaymail.com/news/malaysia/2021/01/20/covid-19-vaccine-panel-chief-urges-malaysians-not-to-panic-over-norwegian-d/1942343
- https://www.nst.com.my/news/nation/2021/01/657965/covid-19-vaccine-timeline-allows-malaysia-study-efficacy-and-safety
- https://machinelearningmastery.com/calculate-bleu-score-for-text-python/
- https://trec.nist.gov/data/reuters/reuters.html
- Khan, R., Qian, Y., & Naeem, S. (2019). Extractive based Text Summarization Using K-Means and TF-IDF. International Journal of Information Engineering & Electronic Business, 11(3).
- Haider, M. M., Hossin, M. A., Mahi, H. R., & Arif, H. (2020, June). Automatic Text Summarization Using Gensim Word2Vec and K-Means Clustering Algorithm. In 2020 IEEE Region 10 Symposium (TENSYMP) (pp. 283-286). IEEE.
- Kågebäck, M., Mogren, O., Tahmasebi, N., & Dubhashi, D. (2014, April). Extractive summarization using continuous vector space models. In Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC) (pp. 31-39).

Thank You.