IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

Automatic Text Summarization (ATS) By Using Recursive Neural Networks

K.TULASI KRISHNA KUMAR [*1], ADARI KARTHIK [*2]

[*1]Associate Professor, Department of Computer Science & Engineering,
[*2]MCA Student, Department of Computer Science and Application, Sanketika Vidhya Parishad
Engineering College, P.M. Palem, Visakhapatnam, Andhra Pradesh, 530041

ABSTRACT

Digital papers containing textual information quickly add together to produce enormous amounts of data. Most of these papers are unstructured, meaning that they are free-form text that has not been arranged into conventional databases. As a result, processing documents is a menial task, largely because of a lack of standards. Implementing computerised text analysis tasks has consequently grown to be incredibly challenging. In order to comprehend this ever-growing, challenging-to-handle amount of information, Automated Text Summarization (ATS) can aid by compressing the text while keeping relevant information. Recursive neural network methods are what I'm employing in this case for ATS. The author first summarises the current state of the art before going into detail on the primary issues with ATS, as well as the challenges and community-provided remedies. The paper discusses recent developments in ATS as well as present-day uses and fashions. The methods used are symbolic, linguistic, and statistical. Also, a number of examples are provided to help clarify the theoretical ideas.

KEY WORDS:

Automated Text Summarization, Recursive Neural Network, Conventional Databases, Linguistic, Statistical.

1. INTRODUCTION

Text mining, which employs various Artificial Intelligence methods, is also known as text analysis. Text mining is the process of extracting high-quality information from unstructured data by creating patterns and locating crucial keywords. Among the tasks associated with text mining are text classification, which involves categorizing the text according to factors like genre, emotional analysis, which reveals the tone in which an author's phrases were written, and sentiment detection. Document clustering is the process of grouping documents together using unsupervised learning and text summarizing to produce a concise and accurate text. Each task differs from the others and has its own methodology and investigation. The goal is to extract useful numerical indices from the text from the unstructured material. Make the text's information accessible to the various algorithms as a result. The documents' information can be extracted to create summaries. As a result, you can examine individual words and word groups in texts. Text mining, to put it simply, "turns text into numbers." such involves the use of unsupervised learning techniques in predictive data mining initiatives.Condensing a lengthy text into a manageable length while maintaining the essential informational components and the meaning of the content is known as summarization. Since manually summarizing material requires a lot of time and is generally difficult, automating the process is becoming more and more popular, which is a major driving force behind academic research.

Text summarization has significant uses in a variety of NLP-related activities, including text classification, question answering, summarizing legal texts, summarizing news, and creating headlines. Moreover, these systems can incorporate the creation of summaries as a preliminary step, which shortens the text. Word processing techniques like word sequence prediction can help you create text summaries by using fewer keystrokes when you need to type words. Hence, transcribing and summarizing a lengthy written document is a laborious and time-consuming operation. With the vast and enormous amount of data that is being exchanged in this digital data environment, it is essential to develop machine learning algorithms that can produce accurate summaries automatically.

2. LITERATURE SURVEY

In this section we will mainly discuss about the background work that is carried out in order to prove the performance of our proposed Method. Literature survey is the most important step in software development process. For any software or application development, this step plays a very crucial role by determining the several factors like time, money, effort, lines of code and company strength. Once all these several factors are satisfied, then we need to determine which operating system and language used for developing the application. Once the programmers start building the application, they will first observe what are the pre-defined inventions that are done on same concept and then they will try to design the proposed text summarizing task in some innovated manner.

MOTIVATION

[1]." Summarization method using Fuzzy rules" by Fabio Bif Goularte

Because of the vast amount of online data and the potential for the text summarization task to extract important information and knowledge in a way that could be easily handled by humans and used for a variety of purposes, including expert systems for text assessment, the task has gained much importance. In order to identify the most crucial information in the assessed texts, this research provides an automatic text assessment procedure that uses fuzzy rules on a range of extracted attributes. These texts' automated summaries are contrasted with reference summaries written by subject-matter specialists. In contrast to earlier ideas in the literature, our method reduces dimensionality and, as a result, the number of fuzzy rules needed to summarise text by looking at linked aspects.

[2]." Deep learning approaches for summarization of legal documents" by Deepa Anand

The author suggested a strategy for condensing the judicial judgment documents from India, and the author employed a semi-supervised neural network methodology to achieve it. Legal decision records being accessible in digital form presents many potential for information extraction and application. Due to their distinctive structure and high level of complexity, automatic summarizing of these legal writings is both essential and difficult. For greater success, earlier efforts in this manner have concentrated on a specific sub-domain and used large labeled datasets, hand-engineered features, and domain knowledge. In this paper, we present simple general neural network-based algorithms for the task of summarizing Indian court decision papers. For this objective, we investigate two neural network designs.

[3]." Summarization of multi-documents using RNN " by Jingqiang

A multi-model extractive summarizer based on neural networks was proposed. Multi-modal summarization is required due to the Internet's quick increase in multi-modal documents with images. Current developments in neural-based text summarising demonstrate the effectiveness of deep learning as a summary technique. This study suggests a multi-modal RNN-based extractive neural-based multi-modal summarization approach. This technique initially uses a multi-modal RNN to encode documents and images, and it then uses a logistic classifier with features for text coverage, text redundancy, and image set coverage to determine the likelihood that sentences will be summarised. We add to the Daily Mail corpora by gathering online pictures. Experiments demonstrate that our approach outperforms cutting-edge neural summarization techniques. In contrast to cutting-edge neural summary techniques, we developed a neural multi-modal extractive summarization method.

3. EXISTING METHODOLOGY

There was no appropriate way to summarize the text automatically in the existing system. In the existing system there was no proper method to summarize the text from a given input paragraph or file by using any pre-defined data mining algorithms. All the existing methods try to follow manual approach which is somewhat complicated to summarize the text messages. The following are the main limitations in the existing system.

LIMITATION OF EXISTING SYSTEM

1. More Time Delay in finding the important text in appropriate area.
2. There is no accurate summarization by using manual approach/
3. This is not efficient method to summarize the text automatically.
4. All the primitive methods use manual approach and hence lot of manpower required to extract the information.
5. There is no optimization of time for extracting the text summarization.

4. PROPOSED SYSTEM & ITS ADVANTAGES

Here, we've spoken about the machine learning strategy that creates summaries of articles of any length using artificial neural networks. Particularly, it has been discovered that when used to tackle the challenge of text summarization, the Encoder-Decoder recurrent neural network (RNN) architecture created for machine translation produces promising results. By using the proposed system we can easily summarize the text from large sentences or paragraphs or from a text book.

ADVANTAGES OF PROPOSED SYSTEM:

The following are the benefits of the proposed system. They are:

1) By using RNN Model, we can easily summarize the text messages.

2) It is simple to construct because we are using parallel processing here, with an encoder taking the input sequence and producing a vector output and a decoder using the previous vector output as an input and producing the final output sequence.

3) Result from analysis clearly state that the RNN gives best result in very less time.

5. PROPOSED ALGORITHMS

The following RNN model was used for text summarizatio are deployed in this current application. They are as follows:

Recurrent Neural Network (RNN):

RNN operates on the tenet that each layer's output is saved and fed back into the system's input in order to forecast that layer's output. How to change a feed-forward neural network into a recurrent neural network is described below:

To create a single layer of recurrent neural networks, the nodes from several layers of the neural network are compressed. The network's parameters are A, B, and C.

There were a few problems with the feed-forward neural network, which led to the development of RNN:

1) Cannot process sequential data;
2) Only takes into account current input;
3) Unable to memorized prior inputs

The RNN offers a remedy for these problems. An RNN can handle sequential data, accepting both the input data being used at the moment and inputs from the past**.**

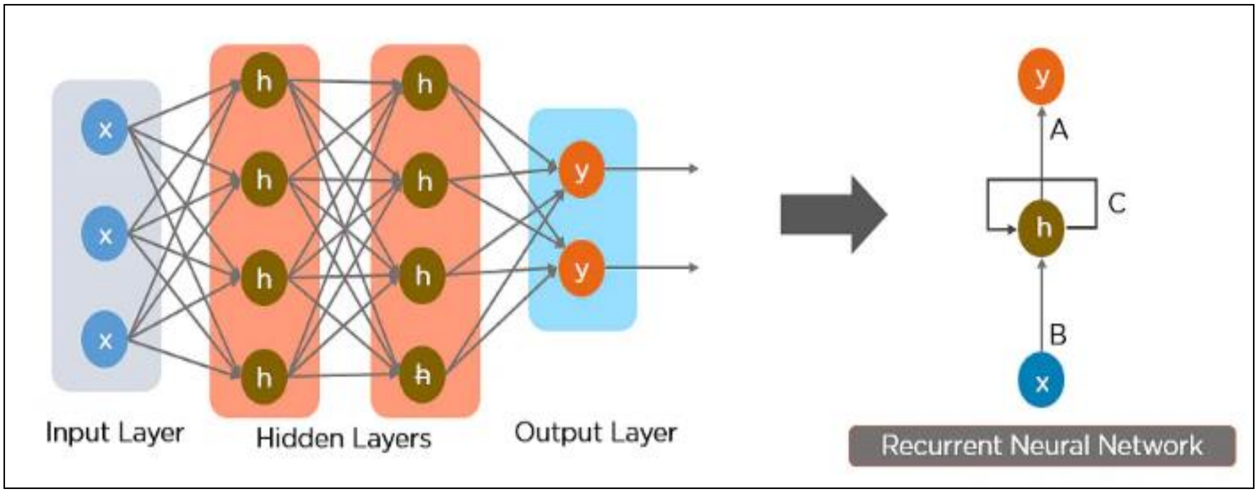


Figure 1. Represents the Basic RNN Model

Applications of Recurrent Neural Networks

The following are the applications of RNN used in several areas. They are as follows:

1. **Image Captioning**

RNNs are used to caption an image by analyzing the activities present.

2. **Time Series Prediction**

Any time series problem, like predicting the prices of stocks in a particular month, can be solved using an RNN.

3. **Natural Language Processing**

Text mining and Sentiment analysis can be carried out using an RNN for Natural Language Processing (NLP).

4. **Machine Translation**

Given an input in one language, RNNs can be used to translate the input into different languages as output.

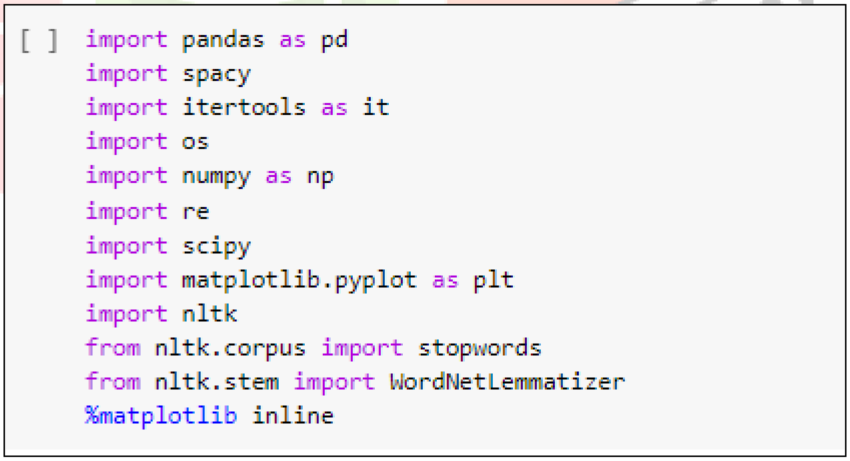
6. IMPLEMENTATION PHASE

The step of implementation is when the theoretical design is translated into a programmatically-based approach. The application will be divided into a number of components at this point and then programmed for deployment. The front end of the application takes Google Collaboratory and as a Back-End Data base we took Daily Caller. csv as dataset. Python is being used in this instance to implement the present application. The following 5 modules make up the bulk of the application. They are listed below.:

1. Import Necessary Libraries
2. Load Dataset Module
3. Data Pre-Processing
4. RNN Text Generator
5. Model Selector

1) IMPORT NECESSARY LIBRARIES

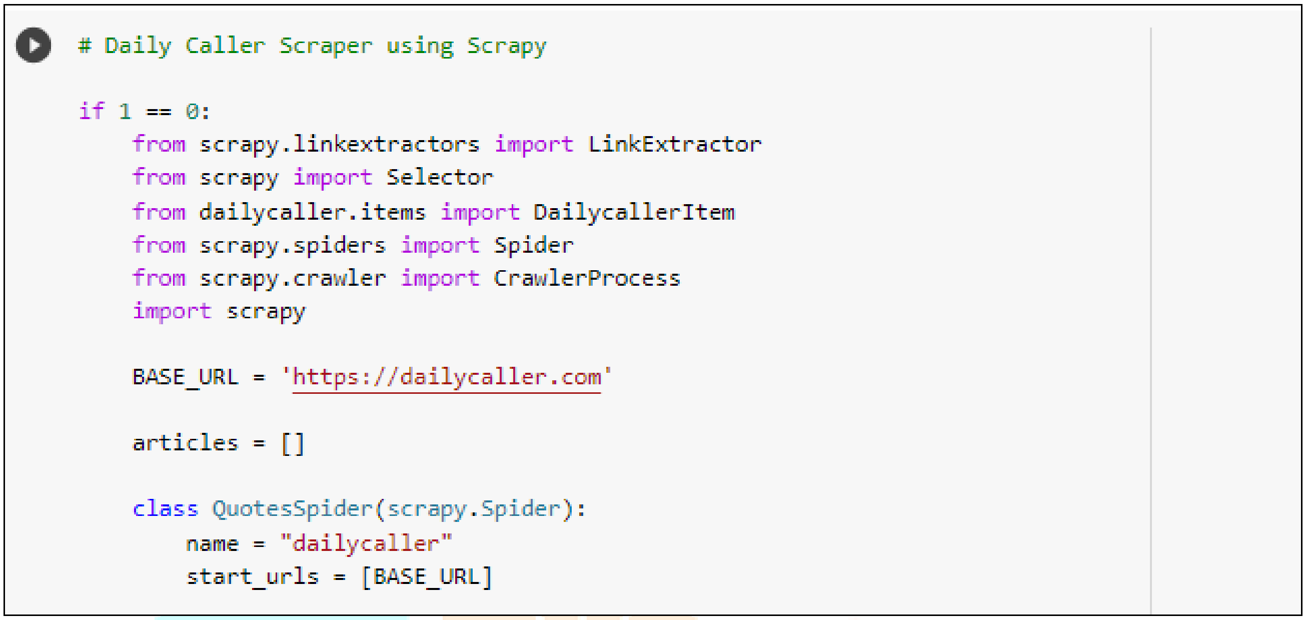
We must first import all the relevant libraries into this module in order to create the model. Here, we make an effort to employ every library available for converting data in a useful way. We try to import the numpy module since the data is separated into numerical values that the system can quickly identify, and we use the matplot library to plot the data in graphs and charts.



```
[ ] import pandas as pd
    import spacy
    import itertools as it
    import os
    import numpy as np
    import re
    import scipy
    import matplotlib.pyplot as plt
    import nltk
    from nltk.corpus import stopwords
    from nltk.stem import WordNetLemmatizer
    %matplotlib inline
```

2) LOAD DATASET MODULE

We attempt to load the dataset that has been downloaded or gathered from the Google repository in this module. The dataset is described as follows.



```
# Daily Caller Scraper using Scrapy

if 1 == 0:
    from scrapy.linkextractors import LinkExtractor
    from scrapy import Selector
    from dailycaller.items import DailycallerItem
    from scrapy.spiders import Spider
    from scrapy.crawler import CrawlerProcess
    import scrapy

    BASE_URL = 'https://dailycaller.com'

    articles = []

    class QuotesSpider(scrapy.Spider):
        name = "dailycaller"
        start_urls = [BASE_URL]
```

The dataset is formed as daily caller scraper using scrapy module.

3) DATA PRE-PROCESSING MODULE

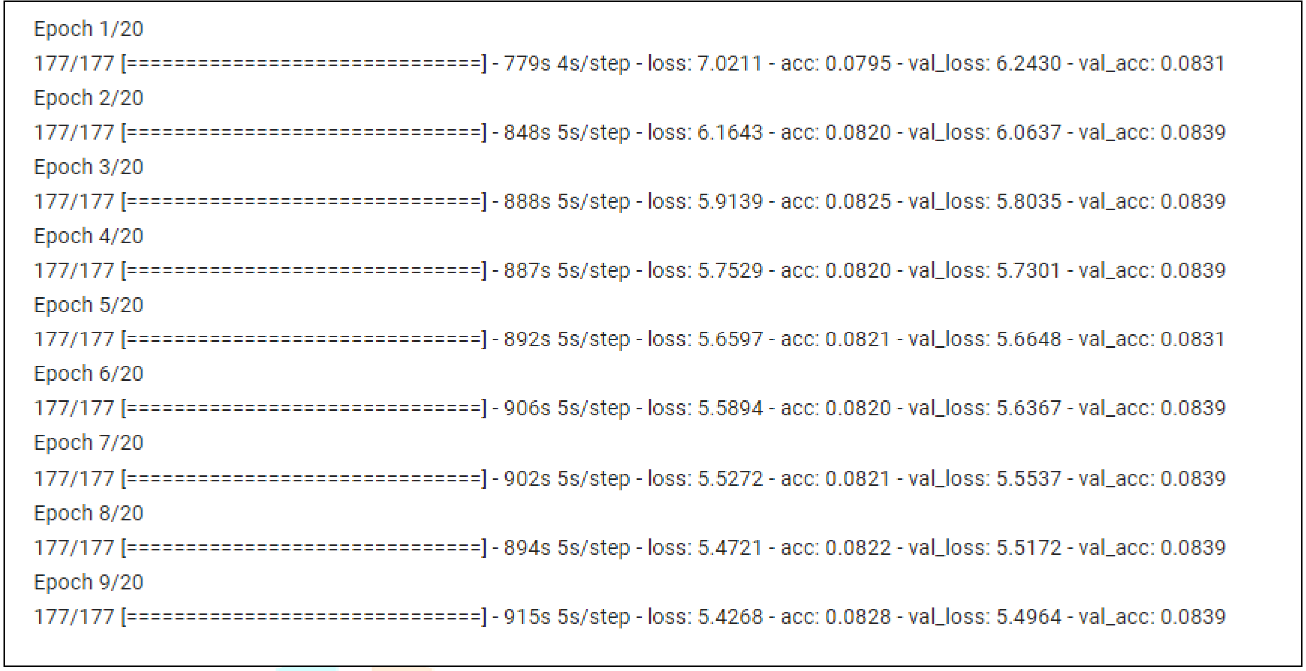
In this part, we try to perform a pre-processing operation on the incoming dataset to identify any missing values or incomplete data. The programme will load only valid rows that have all the valid inputs if there are any such data present in the dataset.

4) RNN TEXT GENERATOR

Creates a text dataset contains the one-hot encoded text data. It produces batches of sequences of encoded labels. We split the text data into batches are used to train the RNN, and we sample a random chuck of the text (with given length) to evaluate the performance of our model.

5) MODEL SELECTOR

Performs randomized search and rank the models by accuracy. It selects the best ranking models and allows lengthy searching (for hours/days).



```
Epoch 1/20
177/177 [==============================] - 779s 4s/step - loss: 7.0211 - acc: 0.0795 - val_loss: 6.2430 - val_acc: 0.0831
Epoch 2/20
177/177 [==============================] - 848s 5s/step - loss: 6.1643 - acc: 0.0820 - val_loss: 6.0637 - val_acc: 0.0839
Epoch 3/20
177/177 [==============================] - 888s 5s/step - loss: 5.9139 - acc: 0.0825 - val_loss: 5.8035 - val_acc: 0.0839
Epoch 4/20
177/177 [==============================] - 887s 5s/step - loss: 5.7529 - acc: 0.0820 - val_loss: 5.7301 - val_acc: 0.0839
Epoch 5/20
177/177 [==============================] - 892s 5s/step - loss: 5.6597 - acc: 0.0821 - val_loss: 5.6648 - val_acc: 0.0831
Epoch 6/20
177/177 [==============================] - 906s 5s/step - loss: 5.5894 - acc: 0.0820 - val_loss: 5.6367 - val_acc: 0.0839
Epoch 7/20
177/177 [==============================] - 902s 5s/step - loss: 5.5272 - acc: 0.0821 - val_loss: 5.5537 - val_acc: 0.0839
Epoch 8/20
177/177 [==============================] - 894s 5s/step - loss: 5.4721 - acc: 0.0822 - val_loss: 5.5172 - val_acc: 0.0839
Epoch 9/20
177/177 [==============================] - 915s 5s/step - loss: 5.4268 - acc: 0.0828 - val_loss: 5.4964 - val_acc: 0.0839
```

Here our model is generating with an accuracy of 83.9 % compared with other primitive models.

7. CONCLUSION

There are numerous tasks involved in producing the précis from the extracted text summary. The author suggested using sentence ranking based on the chosen attributes for the sentences as a statistical method for creating an extracting précis of a sports article. Finding the relevant sentences in the given paper without losing their meaning is the most significant step. Comparing the suggested method to a manual-generated summary and an online summarizer tool, 74% f-measure, 73% precision, and 76% recall on average are obtained. By using an RNN model, we were able to extract text with an accuracy of 83.9%. We intend to expand the same application using the pre-trained Google BERT model in future work and boost the precision of our suggested model.

8. REFERENCES

1. Anand, Deepa & Wagh, Rupali. (2019). Effective Deep Learning Approaches for Summarization of Legal Texts. Journal of King Saud University - Computer and Information Sciences. 10.1016/j.jksuci.2019.11.015.
2. Azar, Mahmood & Hamey, Len. (2016). Text Summarization Using Unsupervised Deep Learning. Expert Systems with Applications. 68. 10.1016/j.eswa.2016.10.017.
3. Goularte, Fábio & Nassar, Silvia & Fileto, Renato & Saggion, Horacio. (2018). A Text Summarization Method based on Fuzzy Rules and applicable to Automated Assessment. Expert Systems with Applications. 115. 10.1016/j.eswa.2018.07.047.
4. https://heartbeat.fritz.ai/the-7-nlp-techniques-that-will-change-how-you-communicate-in- the-future-part-i-f0114b2f0497
5. https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6
6. https://www.upgrad.com/blog/what-is-text-mining-techniques-and-applications/

7. Jain, Aditya & Bhatia, Divij & Thakur, Manish. (2017). Extractive Text Summarization Using Word Vector Embedding. 51-55. 10.1109/MLDS.2017.12.

8. J. Chen and H. Zhuge, "Extractive Text-Image Summarization Using Multi-Modal RNN," 2018 14th International Conference on Semantics, Knowledge and Grids (SKG), Guangzhou, China, 2018, pp. 245-248.doi: 10.1109/SKG.2018.00033

9. J. N. Madhuri and R. Ganesh Kumar, "Extractive Text Summarization Using Sentence Ranking," 2019 International Conference on Data Science and Communication (IconDSC),

10. Jo, Duke Taeho. (2017). K nearest neighbor for text summarization using feature similarity. 1-5. 10.1109/ICCCCEE.2017.7866705.

11. N. S. Shirwandkar and S. Kulkarni, "Extractive Text Summarization Using Deep Learning," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2018, pp. 1-5. doi: 10.1109/ICCUBEA.2018.8697465

12. P. Krishnaveni and S. R. Balasundaram, "Automatic text summarization by local scoring and ranking for improving coherence," 2017 International Conference on Computing Methodologies and Communication (ICCMC), Erode, 2017, pp. 59-64. doi: 10.1109/ICCMC.2017.8282539

13. Valverde Tohalino, Jorge & Amancio, Diego. (2017). Extractive Multi-document Summarization Using Multilayer Networks. Physica A: Statistical Mechanics and its Applications. 503. 10.1016/j.physa.2018.03.013.

9. About the Authors

K. TULASI KRISHNA KUMAR is currently working as Associate Professor in Department of Computer Science and Engineering at Sanketika Vidhya Parishad Engineering College, P.M. Palem, Visakhapatnam, Andhra Pradesh. He has more than 13 years of teaching experience. His research interest includes AI, Java and Python.

ADARI KARTHIK is currently pursuing his 2 years MCA in Department of Computer Science and Applications at Sanketika Vidhya Parishad Engineering College, P.M. Palem, Visakhapatnam, Andhra Pradesh. His area of interest includes C, C++, Java and Python.