IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 9, Issue 3, March 2020

Document Summarizing AI System

Vijay. M.N[1], Vignesh. R.R[2], Sivaprasath. V[3], Tamilalakan. S[4], Priyadharshini. M[5*]

Computer Science and Engineering, KPRIET, Coimbatore, India[1-5]

*Corresponding author

Abstract: Creating short summaries of documents is obtaining salient information from an authentic text document. The extracted information is attained as a summarized report and consulted as a concise summary to the user. It is very crucial for us to understand and to describe the content of the text. The extractive summarization technique focuses on choosing how paragraphs, essential sentences, etc., creates the original documents in precise form and presents a summary that only contains parts of the original document. The efficiency of summarization resides in having identifying and presenting the key entities in the document. The proposed system aims at creating an extractive summary of multiple documents and enables us to find the relevance of the contents in those documents. This is enabled with a user interface to pose a query on set of multiple documents and present the most relevant documents in the order. Simple machine learning algorithms are used to perform this and the performance evaluation of the system could help the progress of research activities further to do the same as abstractive summarization using deep neural networks.

Keywords: Summarization, Machine Learning, Tokenization, Algorithms, Spacy.

I. INTRODUCTION

Text Summarization is a technique used to provide shorter summary of a long text. The important points and non-redundant content which are significant among the contents retrieved from the available large pool of texts. In this age of technology, data gets generated every second in the internet. By the end of 2025, the data will grow up to 175 ZB (Zettabytes), estimated by the IDC (International Data Corporation). With such a large amount of data flowing in the internet, there is a need to develop Machine Learning algorithms to summarize the contents. Now-a-days people don't have a time to read all the content in the internet. So summarization plays a major role in the internet by news summarization, definitions of the technical terms, etc., The proposed AI system uses Machine Learning for text summarization [1]. The following are categories of Machine Learning Supervised Learning, Unsupervised Learning, Reinforcement Learning, Semi-supervised Learning, Feature Learning, Self Learning and Sparse Dictionary Learning. Supervised Learning model can be created by using historical data (some set of training data). Unsupervised Learning model can have only input data and by clustering the input data based on the model new patterns are generated. Reinforcement Learning provides trial and error concept by learning itself as what human a does. Tkinter is a framework in python that can be used to create the GUI for the system. By integrating the Tkinter and Machine Learning model the Document Summarizing AI System is created. This proposed AI System provides the data in a concise form by extracting from voluminous amount of data. It provides a unique service that can be used in the News summarization [5], Article Summarization, etc., in future.

II. OBJECTIVE

The objective of the system is to summarize large data. The data could be in any form such as text, PDF and Webpage URL. The summarized content doesn't miss the important facts of the document. The time taken to summarize the content will be faster. It could also work on platform independent.

III. TEXT SUMMARIZATION METHODS

(A) SUPERVISED LEARNING

Supervised learning is a technique in which training is given to the machine using data which is well labelled. In Supervised learning input variables x and an output variable Y are used with an algorithm to learn the mapping function from the input to the output.

$$Y = f(X)$$

The aim is to estimate the mapping function so accurate that when you have new input data (x) that you can predict the

DOI 10.17148/IJARCCE.2020.9327

output variables (Y) for that data. The Machine learning model that learns from the old input data and makes new prediction as output is called supervised learning.

Naive Bayes

Naïve Bayes uses the approach of machine learning, the model trains the classifier and predicts the output based on the calculation of singular-value decomposition (SVD). Before training the model, it needs two concepts of recursive feature elimination and SVD-feature ranking [4]. In this method, the training dataset is used as reference and the summarization process is modelled as a classification problem: sentences are divided as summary sentences and non-summary sentences based on the features that they possess. The classification probabilities are learnt statistically from the training data, using Bayes' rule: where, s is a sentence from the document collection, F1, F2, F3...FN are characteristics used in classification. S is the summary to be generated and P (s∈< S | F1, F2, F3..., FN) is the probability that sentence s will be chosen to form the summary given that it possesses features F1, F2, F3…., FN.

$$\mathrm{P}(A|\mathrm{B}) = \frac{\mathrm{P}(\mathrm{B}|\mathrm{A})\mathrm{P}(\mathrm{A})}{\mathrm{P}(\mathrm{B})}$$

(B) REINFORCEMENT LEARNING

Reinforcement learning can think and act like human brain. This Reinforcement Learning is about selecting a suitable action to maximize reward for a particular situation. It is designed by various algorithms to find the best possible behaviour or path it should take in a specific situation. It is a trial and error. Without the training dataset, it is forced to learn from its experience. Mostly, it is seen in robotics.

Markov Decision Process

The mathematical approach for mapping a solution in reinforcement Learning is recon as a Markov Decision Process or (MDP). This approach can also be called as a discrete time stochastic control process. It can be applied in various fields such as information engineering, production, and economics. By using this approach, we can find the best possible solutions of complex problems that can be solved by using dynamic programming and reinforcement learning.

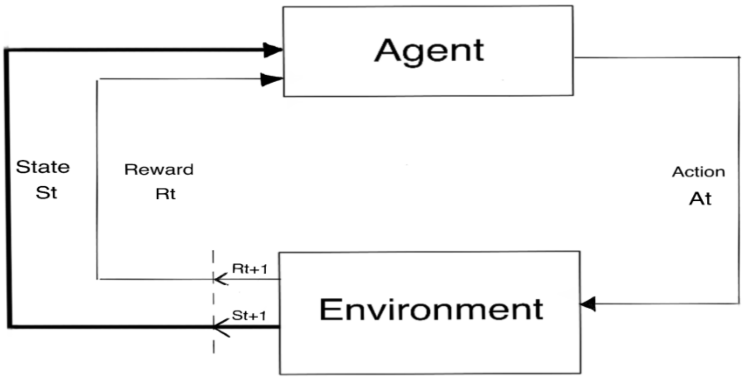

Fig. 1 Markov Decision Process

(C) UNSUPERVISED LEARNING

Unsupervised learning is a type of machine learning algorithm used to draw patterns from input sets. Cluster analysis is the most common method to identify the hidden patterns or grouping in data. The cluster model identifies the similar index by measuring the Euclidean or probabilistic distance self-organizing maps uses neural networks that learn the topology and distribution of the data. In Hidden Markov models, Observed data is used to recover the sequence of states.

K-Means Clustering

In unsupervised learning k-means is one of the simplest algorithms that solve the well-known clustering problem. This algorithm simply classifies the given data into a number of clusters (assume k clusters) which is to be given prior. The main idea is to define k centres, one for each cluster. These centroids should be placed in a correct way because different location causes different result. So, placing them as much as far possibly will improve the results. The next step from a given data takes each point and associates it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done.

DOI 10.17148/IJARCCE.2020.9327

$$J = \sum_{j-1}^{k}\sum_{i-1}^{n}\left\|X_{i-1}^{(j)} - C_j\right\|^2$$

IV. PROPOSED SYSTEM

(A) SYSTEM ARCHITECTURE

This architecture diagram named as Fig. 2 represents the core functioning of the AI system. All inputs are converted into text format and pushed into the summarizer module. The contents from the web URL are fetched using web scraping frameworks such as BeautifulSoup. In this module, it first identifies the stopwords and removes that to reduce the unwanted number of words. Then it determines the word frequency count for each word in the matrix representation and stores the maximum frequency words in the word frequency list. Word frequency can be calculated by frequency of each word to the maximum frequency.Then the sentences can be tokenized in the format of array of lists. Using sentence score method to calculate score for each sentences and highest ranked sentences will be returned from the summarization module and results will be displayed in the output console.

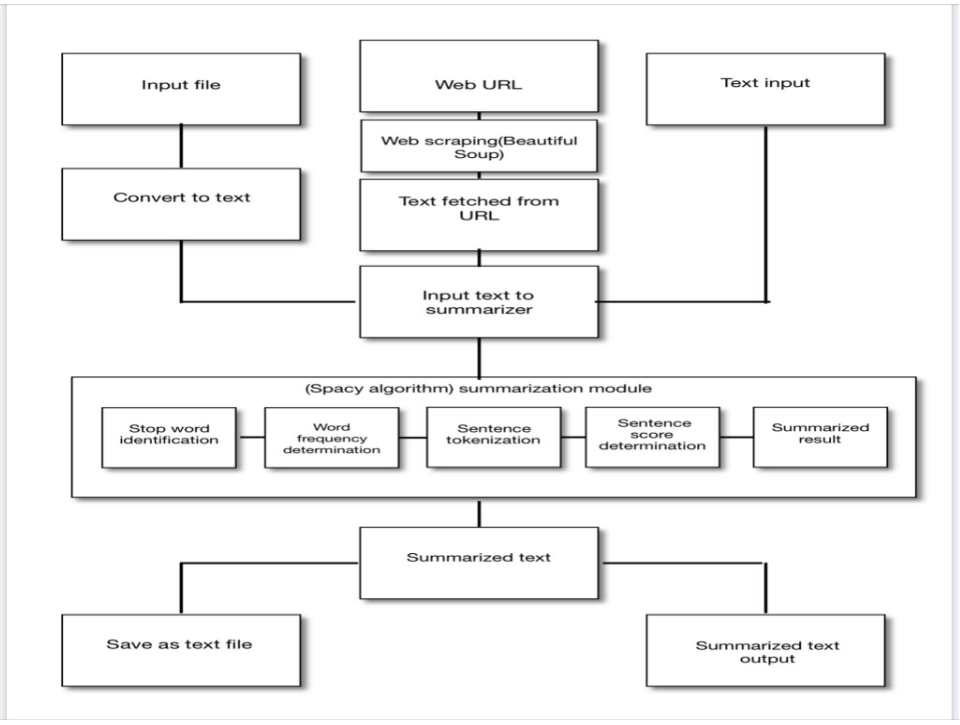

Fig. 2 System Architecture

Input Module

The input to the Document Summarizing AI System is the textual data. There are three modes of input text as follows.

1. Raw Text
2. Text File
3. Web URL

The text from the text file is read using file operations. The text from the web URL is scraped using the BeautifulSoup which is a web scraping framework [2]. The URL is opened using the url lib library. The raw text and other inputs which are converted to text are given to the summarization module as input.

Summarization Module

The summarization module receives the text as input. It has few components in it to produce a summarized text as follows.

1. Stop Word Identification
2. Word Frequency Determination
3. Sentence Tokenization
4. Sentence Score Determination
5. Summarized Result

The text has a lot of stop words and it needs to be eliminated because it is not needed when summarizing a text. The python packages have a list of stop words and those stop words from these text are removed. Then the word frequency needs to be determined. A dictionary is created where the words are keys and whose values are updated with the no of times it occurs in the text. Then the given text contains sentences and those sentences are tokenized. Then based on the frequency of the words the sentence scores are determined. Thus sentences are tokenized and their scores are calculated. Then based on the score of the sentence the summaries are produced. The sentence having the maximum score are considered for summarizing the text. Then the summarized result is given as output to the output window.

Results

The summarized text is obtained as output from the summarization module. Then the output is made available to the users in two modes and they are as follows.

1. Summarized Text Output in the output window
2. Summarized Text File

The summarized text is displayed in the output window to the user as output. Then the summarized text can be saved as a file. We use file operations to write the summarized text output to the files. The files are named based on the document no, date and time.

TABLE I Comparison of Algorithms

ALGORITHM	**ORIGINAL TEXT** (in number of words)	**SUMMARIZED TEXT** (in number of words)
SPACY	516	140
NLTK	516	145
GENSIM	516	168
SUMY	516	155

From the above table named as Table I, we observe that there are four algorithms which are compared based on their summarization. A text of length 516 words is considered for summarization. The summarization is done using four algorithms and each algorithm produces a summarized text. Among the four algorithms we found Spacy is best which produces a shorter summarized text of length 140 words and it is meaningful as well.

(B) LIBRARIES USED IN THE AI SYSTEM

Spacy

Advanced in NLP (Natural language processing) [3] written in python and cython (Open-source library). It is specifically designed for production usage software and used to build this NL (Natural Language) Artificial System. Spacy offers some features are independent, more flexible than other statistical models. It provides a variety of linguistic annotations to give you text’s grammatical structure from insights. This AI system uses unsupervised Learning with the help of spacy.

Tkinter

It is the standard library (GUI) for python used as a front end applet provider in the Document Summarizing AI System. Tkinter provide a powerful OO (object-oriented) interface to Tk (tkinter) GUI tk (toolkit) and include with standard Microsoft windows, Linux, mac installs of python. Tk (tkinter) provides text boxes, buttons, labels (In GUI application) and these are commonly called as widgets which can be implemented as a front end applet in the AI system.

Tokenizer

In the AI system using tokenizer to split large contents into smaller parts like sentences to words, paragraphs to sentences such as words, keywords, phrases, symbols and other elements called tokens. Two types of tokenizer can be used, one is tokenizer for words and other is tokenizer for sentences. Some characters like punctuation are removed. In parsing and text mining have input from tokens. In a words or sentence tokenizer break text into tokens whenever sees any whitespace. Finally, the tokenized content will be stored in a separate list.

V. CONCLUSION

This project aims at converting the larger text content into a shorter summarized text which contains the important and meaningful information. This project helps the students and users to summarize the websites. So their surfing time will be less and the learning time will get improved. Adopting efficient methodologies in the day to day operations will

DOI 10.17148/IJARCCE.2020.9327

increase their productivity. We can focus more on important aspect on task while using efficient methodologies. It helps us to look for more meaningful information amidst the very large collection of contents in the Internet which will take immense time to read all of them. The time taken to document the summarized text will be off the picture since we have the option to save the summarized text as a file. We found Spacy algorithm is best because it produces shorter and meaningful summaries.

VI. FUTURE WORK

In future we improve the efficiency of the system and it can be very useful in the field of automatic news summarization. It can also use in the API to provide search results in a summarized form. The accuracy of the algorithms can be improved based on the understanding of the domain specific contents. The future advancements in the NLP which is a research area will allow us to create summaries which are more like the words delivered by a human. It will be grammatically correct and meaningful at the same time. We can extend this functionality to various file formats having text content. We can improve the quality using neural networks.

REFERENCES

[1]. A Survey on Extractive Text Summarization, Web page at https://www.researchgate.net/publication/317420253_A_survey_on_extractive_text_summarization consultancy-services/cloudcomputing/how-is-cloud-computing- different-from-traditional-itinfrastructure/ . Date of published: Jan 2017.
[2]. Summarizing Websites Automatically, Web page at https://link.springer.com/chapter/10.1007/3-540-44886-1_22.
[3]. Text Summarizing using Natural Language Processing. at http://web.cs.wpi.edu/~claypool/mqp/sv/2018/juniper/juniperfinal.pdf. Date of published : March 2018.
[4]. Explorer Neural Latent Extractive Document Summarization, https://www.semanticscholar.org/paper/Explorer-Neural-Latent-
[5]. Extractive-Document-Lapata/16d0afaeb8419ec1c37c3473ab581df916148d72S, Date of published:2018.
[6]. Automatic News Summarization with sentence vector Offset, https://ieeexplore.ieee.org/document/8924017. Date of published: 2019.

DOI 10.17148/IJARCCE.2020.9327