Global Journal of Modeling and Intelligent Computing (GJMIC)

Vol. 2, No. 1, January, 2022 ISSN: 2767-1917

STATISTICAL APPROACH FOR EXTRACTIVE TEXT SUMMARIZATION

Vasudev Sharma, Thapar Institute of Engineering and Technology, India (vsharma13.1998@gmail.com)
Jasmeet Singh, Thapar Institute of Engineering and Technology, India (jasmeet.singh@gmail.com)

ABSTRACT

The extractive text summarisation technique is used for the extraction of important points of documents by using a subset of the sentences present in the original document. The sentences of the documents are extracted and are given a score. The model for text summarisation is created by using the sentences extracted and their respective scores. The sentences of the document are arranged according to their score. The model created is then used for giving out summaries and the final results of the text summariser are evaluated using metrics to measure the accuracy of the model. The model is created using the statistical techniques. The text summarisation problems falls into the category of Natural Language Processing which is concerned with the interaction between computers and the human languages. Once the summary of the document is outputted using evaluation parameters such as precision, recall and F-score we find out how much the summary differentiates from that actual summary that was manually created.

Keywords: - Extractive Text Summarisation, Natural Language Processing, Gisting Evaluation, Text Frequency-Inverse Document Frequency, Natural Language Tool Kit.

1. INTRODUCTION

Text summarisation (Das et al., 2007 & Ferreira et al., 2013) is the process of creating a summary consisting of important points from the primary set of documents which conveys the entire tone of the documents using various techniques so that the documents can be presented in a much more readable format and compact manner in a short span of time. In the past several years there has been a large increment in the amount of data being generated from businesses, science, engineering, social media and various other fields on a daily basis, the collection of document can have size as large as Terabytes or Petabytes. The easy availability of large collections of documents through the internet has made it easy to access them but made it very difficult to manually go through the collection of documents hence automated text summarisers are used for creating a summary consisting of the important points such that these points can give out most amount of the information about the documents saving valuable time, money and preserving its information content and overall meaning (Mohd et al., 2019). The summary also allows the user to focus on what is important since the original collection of documents consists of large set of text and only a certain amount of information is required hence a lot of the unnecessary information needs to be discarded and the desired information needs to be displayed to the user which can prove to be a difficult task while distinguishing the two sets of information, but if successfully implemented it allows the user to focus on only what is necessary. The tool for creating the summary is called an automatic text summariser and the process is called text summarisation. The text summariser can work on a single document which is called single document classification and it can work on multiple documents as well which is called mute-document classification. The type of summary provided by the summariser can be of two types which are generic or query based. The generic summary consists of a general summary of the document whereas the query based summary satisfies a user based query.

The extractive summarisation technique (Gupta et al., 2014, Mehta et al., 2018 &, Saranyamol et al., 2014) aims to produce the summary composed of the subset of the most relevant sentences present in the input documents. The basic architecture of the summariser includes content selection followed by information ordering and then sentence realization, finally we obtain the summary of the document. The statistical approach has been used to create the summariser. The collection of documents used for making the text summariser is called the DUC dataset, which is an annotated collection of documents hence making the entire process multi-document based summarisation. The sentences then are extracted and pre-processed using linguistic techniques (Gambhir et al., 2017) such as removal of stop word, punctuations and sentence segmentation. The sentences are arranged

according to their rank which has been calculated using their statistical features. The model then outputs the summary of the documents. The summary shows the top most percentage of sentences as selected by the user. The aim of the present proposed work is to create a statistical based text summariser capable of text extraction without having any dependency on the language of the text and annotation of the corpora.

2. RELATED WORK/LITERATURE SURVEY

The exponential growth of the internet and advancement in digital technology over the past decade or two has inundated data on web and its overwhelming availability is widespread. The huge volume of information available losses it feasibility of efficient use unless automatic methods to understand, index, classify, clear and concise way of availability to user is not there. The method should save time and resources. Text summarisation technique (Moratanch et al., 2017) generates a compressed version of one or more documents and attempting to give meaning to the document. Text summarisation is of prime importance due to its application to wide fields such as summaries of books, sporting event highlights, stock markets etc. Data in structured and semi structured form usually organized in the form of spread sheets, tables, databases and maps etc. has become critical. These data sets have been published and used by government, social networking sites and other companies for improving services, framing public policies, improve business models and make well informed decisions (Koesten et al., 2020). Google used schema.org markup language to index data sets, documents, images and products (Noy et al., 2019).

The automatic text summarisers can be broadly divided into two broad categories called the abstractive text summarisation and the extractive text summarisation (Tohalino et al., 2018). The abstractive text summarisation involves paraphrasing sections of the source document and requires natural language generation tools and may reuse clauses and phrases from original document. Its creation is more difficult and complex task as extractive summarisation involves concatenation of several sentences which may be selected without modification. Summaries are either generic or query-focused and summarisation task can be supervised or unsupervised. Training data set is needed in a supervised system and unsupervised systems do not use any training data for they generate the summary by accessing only the target documents. Summary can be based on input, output content, details purpose, language (Nazari et al., 2019) and summaries can be indicative and informative summaries. The main aim of the text summarisation is improving the quality of the produced summary using different methods.

A. Statistical Based Approach

Statistical based approach aims to extract information from the input documents using statistical features. This allows the summariser to be language independent and also do not require any sort of annotated corpora. There exists several different kinds of statistical features such as the TF-IDF, Cue phrases, title words, sentence location etc. The features serve as weights which have been assigned to the sentences, a higher weight indicates a better rank. The summary of the document consists of subset of sentences which have a good rank or score.

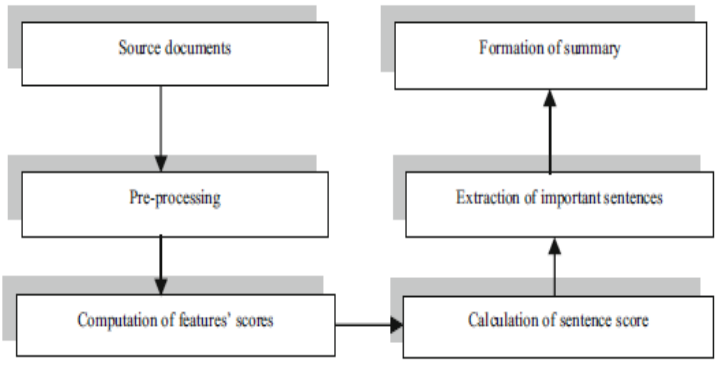

Figure 1: Automatic extractive text summarisation system using statistical techniques (Gambhir et al., 2017)

B. Discourse Based Approach

Discourse based approaches use linguistic techniques for automatic text summarisation. Discourse relations for example cause, contrast and elaboration are considered critical for text interpretation as they indicate how the

sentences are interrelated to form summary. Discourse relations establish rapport between sentences and parts of text. Discourse formalism adduced different resulting structures, namely trees and graph.

C. Topic Based Approach

Topic based approaches as name suggests relate what the document theme (Harabagiu et al., 2005) is about and represented by events occurring in documents. The topic can be represented in different ways such as topic signature, enhanced topic signature, thematic signatures, templates and modeling document content structure.

D. Graph Based Approach

Graph based method can be used for depicting the text structure and relation between sentences by representing the sentences as nodes and relation between sentences as an edge. The method can be used to extract significant, appropriate and informative text in a compressed version. Preprocessing is required in this technique to remove stop words, tokenize sentence etc., followed by ranking of sentences based on importance. The relation between sentences is computed to recognize relevant structure. Finally the sentences are extracted for summary based on their ranking and relevance.

E. Machine Learning Approach

Machine learning approaches are efficient and effective for automatic text summarisation.

1) Naïve Bayes Approach: Naive Bayes approach (Kupiec et al., 1995) is supervised learning method and consider sentence selection as classification problem. Using binary class to determine whether sentence is to be included in summary or not. The features used are word frequency, sentence length, position of paragraph which is responsible for a part of sentence to be part of summary. If S denotes the number of sentences and s denotes a particular sentence with features $F_1, F_2, \dots F_k$,then naïve bayes formula is

$$P(s \in S|F_1, F_2, \dots F_k) = \frac{\prod k_j=1\left(P\left(F_j\middle|s \in S\right)\right)|P(s \in S)}{\prod k_j=1P(F_j)} \quad (1)$$

P (s∈S) is a constant, P= (F_j | s∈S) and P (F_j) can be directly estimated from training set by counting occurrences. P (s∈S | $F_1, F_2, \dots F_k$) represents the probability of the sentences to be included in summary based on the given features possessed by sentence.

2) Artificial Neural Network: Artificial neural network method has been used to select sentences in extractive summarisation (Mutlu et al., 2019).There are three phases of ANN which are training phase, feature incorporation and sentence selection. The training phase identifies the types of sentences to be included in the document summary. A human input is required for the same, the system learns the pattern of summary sentences relation among features is determined, removing common and unimportant features is done then. This ensures important feature stay in summary.

3) Fuzzy Logic Method: Fuzzy logic method (Zadeh, 1965) can use features like similarity to title, keyword, sentence length etc. as input to fuzzy system as knowledge of IF THEN rules are required for summarisation and value 0 to 1 assigned to sentences. The value thus determined the rank of each sentence for final summary.

3. RESEARCH OBJECTIVES/AIM OF THE STUDY

The aim of the study is to create an extractive text summariser capable of delivering a summary which includes a subset of sentences from the original document such that the summariser is able to satisfy the following needs:

1) To create an extractive text summariser which is language independent, hence capable of summarising texts written in any language, this is achieved by using statistical methods and features in order to rank the data. The features used include TF-IDF score, cue phrases and heading title, hence making them independent of the language being used in the document. This technique is not possible in machine learning text summarises since those models are dependent on the language and hence the extractive text summariser can work on a wider variety of languages as compared to other models.

2) To summarise unannotated documents hence there is no need to classify the documents or provide labels and values to the various sentences as it can out to be a cumbersome process in which each sentence of the document needs to labelled. The process of manually labeling can be extremely time consuming and the automatic labeling might not be able to provide fully accurate labels hence it makes the entire process very difficult. This also is achieved by using the statistical features that do not require any dependency on the words of the documents.
3) To reduce the complexity of the text summariser the features or attributes of the sentences are statistical based hence they are quite easy to compute and hence take lesser computing time as compare to machine learning based techniques since they require small number of iterations to compute.
4) The text is subjected to preprocessing techniques so that the memory consumption and the noise in the data can are reduced to a minimum.
5) The summariser should be able to summarise multiple documents and should return the desired summary with user defines compression rates.

4. RESEARCH DESIGN

A. Features

The statistical features used to make the entire model are independent of the language used since we do not need to depend on the individual words as features. The various different scores or features used for the ranking of the all the sentences are given as follows:

1) TF-IDF: The term frequency of a word is denoted by f_{ij} which tells us about the number of occurrences of a particular word 'i' in document 'j'. The logarithmic term frequency has been used along with smoothing so that in case the frequency of the word is very high it would not lead to extremely large TF-IDF score. In case f_{ij} is zero the entire logarithmic term is zero, hence the equation for the

$$f_{ij} = \begin{matrix} 1 + log_e f_{if} f_{ij} > 0 \\ 0 f_{ij} = 0 \end{matrix} \quad (2)$$

The inverse document frequency describes that the words occurring in a few documents are much more useful in distinguishing the documents from the remaining textual documents. The total number of documents are represented by N and the total number of documents in which the word 'i' occurs is denoted by N_i. The IDF equation has been subjected to smoothing so that zero division error does not occur and the equation is given as:

$$IDF = 1 + log_e \frac{1+N}{1+N_i} \quad (3)$$

The term frequency and inverse document frequency are multiplied in order to get the TF-IDF score. A high score in TF-IDF is obtained by having a high term frequency and a low document frequency of the term in the set of documents, such words are very useful in distinguishing the documents. The equation of the TF-IDF if given as:

$$TF - IDF = \begin{cases} \left(1 + log_e f_{ij}\right) * \left(1 + log_e \frac{1+N}{1+N_i}\right) f_{ij} > 0 \\ 0 f_{ij} = 0 \end{cases} \quad (4)$$

2) Cue Phrases: The cue phrases are linguistic expressions which are capable of explicitly signaling discourse structure. The cue phrases are heavily dependent on the genre of the document The number of cue phrases in each sentence are added to the weight of the sentence score, hence the score of the sentence increases with increase in number of cue phrases present in that particular sentence. The Fig.2 some of the cue phrases that have been used in the text summariser in order to calculate the score of the sentences:
3) Heading Title: The number of similar words between the head title of each document and the corresponding sentences are added to the total score of the sentence, hence this gives us the information as to how much each sentence is related to the topic.

The score of the sentence is calculated by summating the different features. The i corresponds to the sentence number and n refers to the total number of sentences where each sentence is ranked according to their $totalscore_i$ and the equation for it if given as:

$$totalscore_i = (TF * IDF)_i + Cuephrases_i + Headingtitle_i \quad (5)$$

B. Architecture

The basic architecture of the text summariser consists of following three sections

1) Content selection: It is the process of extracting sentences based upon the usefulness to the user, hence the non-essential textual data is discarded.

2) Information ordering: content selection followed by information ordering in which each of the sentences are ranked according to the score calculated, a better score implies a better rank and that increases the chances that the sentences is present in the summary.

3) Sentences realization: The sentence realization orders the sentences of the summary in orderly manner so that the summary output is coherent in nature.

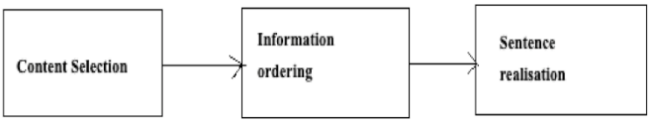

Figure 2: The text summarisation basic architecture

C. Methodology

The statistical based methodology has been used for creating the summariser which has different several phases as given by:

1) The DUC dataset from the year 2004 to 2007 serves as the input to the text summariser. The dataset is collection of unannotated documents consisting of textual data regarding various fields collected in the various different years. The dataset has been written in the XML format hence only certain information is required to create the automatic text summariser. The input documents have the .txt extension.
2) The dataset is then read using the python language in such a manner such that only the heading and text of the document is extracted and the remaining information is discarded. The heading and the text of the documents extracted are stored in difference variables.
3) Once the document texts are obtained they are subjected to sentence level tokenization by using the NLTK library. The tokenizer used in order to break the documents into sentences is called the 'english.pickle' tokenizer.
4) The headings and sentence extracted both are subjected to preprocessing so that linguistic techniques can be applied on them:
    a. All the text is converted to lower case, so that same words in lower and uppers cases are not treated differently which otherwise decrease the accuracy of the summariser and increases the noise in the system.
    b. All the special characters and punctuation marks are removed from the text since they do not add any values to the text summarisation process and take up unnecessary CPU processing time as well as memory.
    c. All the stop words are removed using the NLTK library using its corpus package. The stop words are referred to as the commonly used words in a particular language such words do contain much information about the documents and take additional memory and valuable processing time.
    d. After preprocessing the data the various scores are calculated and the total score of each sentence is obtained. The sentences are ranked according to their scores and the summary of the data is the collection or subset of the sentences with highest score. The number of sentences in the summary depend upon the compression rate as defined by the user.

D. Tools and Technology

The tools and technology required in order to build the extractive text summariser can be divided into two major sub categories that are software and hardware.

1) Software: The software requirements for building the extractive text summariser include:
    a) Python: The Python language is used in order implement the text summariser due to its ability of rapid development of applications and programs. The python language has a simple syntax hence making it easy to code. The Python language consist of extremely rich library such as NLTK, SKLEARN and PANDAS which help us to read and manipulate data in an easy manner.

b) NLTK library: This particular library allows us to process textual data in a very timely manner. This library consists of a collection of libraries and programs for statistical processing for text written in various languages in the python programming language.

c) english.pickle tokenizer: The tokenizer is required to successfully split the document into its respective sentences.

2) Hardware: The hardware used for building the extractive text summariser includes:

a) Central Processing Unit (CPU): The CPU used for running the text summariser is a Quad-Core Intel Core i5 with a clock rate of 3.2 GHz

b) Random Access Memory: The main memory of the computer is an 8 GB 1867 MHz DDR3.

c) Graphic card: The graphic card used is an AMD Radeon R9 M380 with 2 GB memory.

5. RESULTS AND DISCUSSIONS

The extractive text summariser is being evaluated using the precision, recall and F-score metrics. Precision is defined as the ratio of correctly predicted positive observations by the model to the total number of predicted positive observations. The equation for precision is given as follows:

$$Precision = \frac{TP}{(TP+FP)} \tag{6}$$

Recall (Sensitivity) is defined as the ratio of correctly predicted positive observations by the model to the all the observations in the actual class which are positive. The equation for recall is given as follows:

$$Recall(Sensitivity) = \frac{TP}{(TP+FN)} \tag{7}$$

F Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. Intuitively it is not as easy to understand as accuracy, but F is usually more useful than accuracy, especially if you have an uneven class distribution. Accuracy works best if false positives and false negatives have similar cost.

Table 1: Evaluation Metrics Table

Dataset	Average Precision	Average Recall	Average F-score
2004 DUC	0.82	0.76	0.79
2005 DUC	0.79	0.75	0.77
2006 DUC	0.84	0.80	0.82
2007 DUC	0.81	0.79	0.80

True Positives (TP) is the number of classes that are predicted positively and in actuality is also positive True Negatives (TN) is the number of classes that are predicted negatively and in actuality the class is also negative.

Positives (FP) is the number of classes that are actually negative and the predicted class is positive in nature. False Negatives (FN) is the number of classes that are actually positive and the predicted class is negative in nature. Once all the metrics have been calculated their average is taken and they presented in a tabular form as given below.

$$F - Score = \frac{(2*precision*recall)}{(precision+recall)} \tag{8}$$

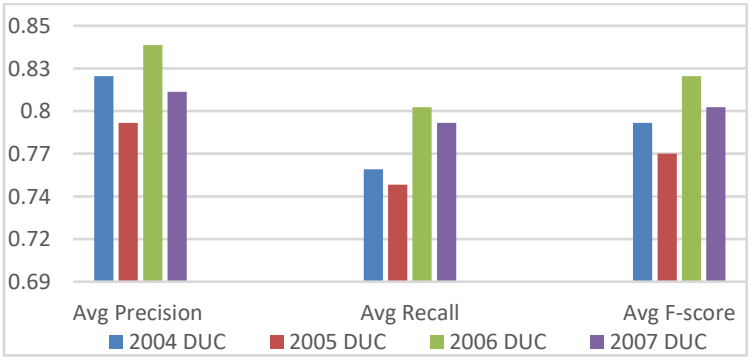

Figure 3: The graphical visualisation of the datasets and there scores.

Once the summary of the document is outputted using evaluation parameters such as precision, recall and F-score help us to distinguish the difference between the actual summary and the summary that was manually created.

6. ANALYSIS

From Table 2, it is clear that precision which illustrates the accuracy of the classifier, is highest for the 2006 DUC dataset, also for the same 2006 DUC dataset, the recall which tells about the actual positives given out by the classifier is highest. F-score which is indicative of the balance of precision and recall and is the primary factor for achieving maximum efficiency of the classifier is highest for same data set. This clearly demonstrates that the 2006 DUC dataset happens to be the optimal option for the classifier.

7. CONCLUSION AND FUTURE SCOPE

The extractive text summarisation is a big research field and has a lot applications. In this particular paper we have described many known extractive text summarisation based including the statistical as well as the machine learning based. The statistical method has allowed us to extract the summary from the DUC dataset with high accuracy. The highest accuracy was presented in the DUC dataset of the year 2006 followed by 2007. The extractive text summariser allows us to extract sentences irrespective of the language being used and with short computation time. The extractive text summariser can be subjected to more pre-processing and various other statistical features can be used in order to increase the accuracy of the system.

Text summarisation is has been a very old field and there exists great interest in this field across the globe due to its vast applications, so the text summarisation continues to improve in order for creating text summarisation approaches or develop efficient summarisation approaches such that summary of higher quality can be generated. The performance of text summarisation in today's world is still moderate and summaries generated are not perfect because they lack consistency and coherency. Therefore the text summarisation system can be made exceptionally good by combining current existing systems with other system so that they can perform better. The extractive text summariser can be subjected to more linguistic techniques and various other statistical and non-statistical features can be used in order to increase the accuracy of the system.

REFERENCES

Das, D. & Martins, A. F. (2007). A Survey on Automatic Text Summarization. Literature Survey for the Language and Statistics II course at CMU, 4, 192-195

Ferreira, R., Cabral, L. de Souza., George, D. L., Cavalcanti, D.C. , Lima, R., Steven, J. S. & Favaro, L. (2013) Assessing Sentence Scoring Techniques for Extractive Text Summarization, Elsevier Ltd., Expert Systems with Applications, 40, 5755-5764.

Gambhir, M. & Gupta, V.(2017). Recent automatic text summarization techniques: a survey. Artificial Intelligence Review, 47,1–66, DOI 10.1007/s10462-016-9475-9

Gupta,V. & Lehal,G.S. (2010) A Survey of Text Summarization Extractive Techniques, Journal of Emerging Technologies in Web Intelligence, 2(3), 258-268

Harabagiu, S. & Lacatusu, F. (2005). Topic themes for multi-document summarization. In: SIGIR' 05: proceedings of the 28th annual international ACM SIGIR conference on research and development in information retrieval, 202–209

Koesten, L. Simperl, E., Blount, T., Kacprzak, E. & Tennison, J. (2020). Everything you always wanted to know about a dataset: studies in data summarization. International Journal of Human-Computer Studies, 135, March, 102367 https://doi.org/10.1016/j.ijhcs.2019.10.004

Kupiec, J., Pedersen, J. & Chen, F (1995). A trainable document summarizer. In SIGIR '95Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval, Seattle, Washington, USA, 68-73

Mehta,P. & Majumder. P, (2018). Effective aggregation of various summarization techniques, Information Processing and Management, 54, 145-158.

Mohd, M., Jan, R. & Shah, M. (2019). Text Document Summarization using Word Embedding Expert Systems With Applications. doi: https://doi.org/10.1016/j.eswa.2019.112958

Moratanch, N. & Chitrakala, S. (2017). A Survey on Extractive Text Summarization. IEEE International Conference on Computer, Communication, and Signal Processing (ICCCSP-2017). DOI: 10.1109/ICCCSP.2017.7944061.

Mutlu, B., Sezer E. A. & Akcayol M. A. (2019). Multi-document extractive text summarization: A comparative assessment on features. Knowledge-Based Systems, 183, 104848 https://doi.org/10.1016/j.knosys.2019.07.019

Nazari,N. & Mahdavi, M. A.(2019). A survey on Automatic Text Summarization. Journal of AI and Data Mining, 7(1), 121-135 DOI: 10.22044/JADM.2018.6139.1726.

Noy, N., Burgess, M. &, Brickley, D. (2019). Google dataset search: Building a search engine for datasets in an open web ecosystem. In: 28th WebConference (WebConf 2019).

Saranyamol, C. S. & Sindhu, L. (2014). A Survey on Automatic Text Summarization, International Journal of Computer Science and Information Technologies, 5(6), 7889-7893.

Tohalino J. V. &. Amancio D. R. (2018). Extractive multi-document summarization using multilayer networks Physica A: Statistical Mechanics and its Applications, 503, 526-539.

Zadeh, L., (1965). Fuzzy Sets. Information and control, 8(3), 338-353.