Extractive Multi-document Text Summarization Leveraging Hybrid Semantic Similarity Measures

Rajesh Bandaru, Dr.Y. Radhika

Department of CSE, GST, GITAM (Deemed to be University)

Visakhapatnam, India

Abstract—Because of the massive amount of textual information accessible today, automated extraction text summarization is one of the most extensively used ways to organize the information. The summarization mechanisms help to extract the important topics of data from a given set of documents. Extractive summarization is one method for providing a representative summary of a text by choosing the most pertinent sentences from the original text. Extractive multi-document text summarization systems' primary goal is to decrease the quantity of textual information in a document collection by concentrating on the most crucial subjects and removing irrelevant material. In the previous research, there are several methods such as term-weighting schemes and similarity metrics used for constructing an automated summary system. There are few studies that look at the performance of combining various Semantic similarity and word weighting techniques in automatic text summarization. We evaluated numerous semantic similarity metrics in extractive multi-document text summarization in this research. In the extractive multi-document text summarization discussed in this research, we looked at numerous semantic similarity metrics. ROUGE metrics have been used to evaluate the model performance in experiments using DUC datasets. Even more, the combination formed by different semantic similarity measures obtained the highest results in comparison with the other models.

Keywords—Extractive text summarization; semantic similarity; sentence scoring; summary

I. Introduction

The amount of data and information available has exploded since the introduction of the World Wide Web. The volume of data has grown to the point where it is nearly difficult for any specific firm to analyze it all, or to summarize it. People are reluctant to engage in reading a lengthy piece of text and, as a result, typically skip crucial sections of it. This has boosted the need for text summarization automation [1].

In general, a person follows the three processes outlined below to create a summary: 1) interpreting the document's content, 2) selecting relevant chunks of meaningful information, 3) putting this content of data. Because of their difficulties, there is limited possibility of automating the first and third processes for any random text. As a result, the majority of techniques aim to automate the second phase [1].

Text summarizing is considered as single or multi-document summary terms of the number of documents studied and summarized at the same time. In single document summarizing, a summary is constructed from a single document, but in multi-document summarization, a series of documents is examined for creating a summary. The task of summarizing many papers is more complex than the process of summarizing a single document. One of the most difficult issues in summarizing many publications is redundancy. Additional classification categories, such as single vs. multi-document categorization and mono-lingual vs. multi-lingual summarization, have been developed in the past based on many other factors [2].

Text summarization techniques often are divided into two categories: abstractive and extractive. The primary goal in extractive summarization included to retrieve the most essential sentences from a document(s) and combine them into a summary. This is in contrast to abstractive summarization, which involves reiterating the information in the text. The extractive summary contains sentences taken directly from the original content, whereas an abstract summary uses terms / expressions not present in the original source. Because of its greater practicality, extractive summarization has become a benchmark in text summarizing [3]. Abstractive text summarizing techniques try to generate summaries that summarize the crux of the text in the same way as people do after studying any text. This employs generative methodologies that can produce meaningful phrases while maintaining the semantics of the source text. This is regarded as a tough topic to tackle, and several novel ways have been presented.

Extractive summarization is divided into three stages: pre-processing, sentence scoring, and sentence selection. Several activities, like as tokenization, phrase and paragraph segmentation, are often carried out during the pre-processing phase. During the sentence scoring step, sentences are ranked based on certain criteria, and every sentence is assigned a score. Finally, the finest sentences are chosen and incorporated in summary during the sentence selection process. As previously stated, one of its most significant issues in multi-document summarizing is duplication, because identical phrases are more likely to be encountered in distinct documents, frequently [4].

Extractive text summarization is a simpler and more reliable method of creating summaries in which key lines from a text are chosen and provided to the user. Each sentence is scored, and the sentences with the highest scores are chosen for inclusion in the extract [5]. This is significantly easier than abstractive summaries, which need the production of phrases and words, as well as their organization into legible sentences, while yet giving an understandable substance of the subject. It

would need a significant amount of natural language processing, making it a significantly more complex task.

With the use of data-driven methods and semantic similarity approaches, extractive summaries will be produced in this study in order to meet the goal of text summarization. This involves analysing the large volume of information and creating a list of the sentences that could be the most helpful and contain the main idea of the text. Although most of the time, people strive to summarize texts in a way that conveys the same sense as the original text and do not see summaries as phrases taken literally from the source [6].

The remainder of the paper can be found in the sections that follow this one: Section 2 presents the results of a survey of the literature on text summarization using various approaches, which was carried out in order to create this paper. After providing a thorough explanation of the proposed algorithm in Section 3, Section 4 presents the results of the experiments conducted as part of the research. After a discussion of the findings and recommendations for future research are in Section 5.

II. RELATED WORK

Since the 1950s, researchers have been studying automatic text summarization. It has since been extensively researched. Researchers working on document summarizing all around the globe are experimenting with a variety of approaches in order to produce ways that deliver the highest results. This work focuses on extractive multi-document summarization.

Various extraction-based strategies for generic multi-document summarization have been suggested so far. Statistical techniques deal with statistical aspects that aid in the extraction of relevant phrases and words from source material. Furthermore, traits and their weights play a significant influence in establishing sentence relevance. This section presents various models employed in the domain of multi-document summarization.

Jesus M. Sanchez-Gomez et al. [5] proposed a model with a set of multi objective functions. The objective functions defined in this work targets to coverage of content and reducing the redundancy. Using a combination of statistical and graph-based methods Mohammad Bidoki et al. [6] proposed a semantic framework for developing an extractive multi-document summarizer system. It is a dialect, unsupervised system. To learn the semantic representation of words from a set of supplied documents, the model uses the word2vec technique. It expands on each phrase using a one-of-a-kind method that employs the most informative and least repetitive words related to the statement's fundamental idea. Phrase expansion implicitly achieves word meaning disambiguation and adapts conceptual density to each sentence's main idea. The importance of sentences is then determined using the graph representation of the documents.

Begum Mutlu and colleagues [7] have created an English dataset including the proceedings of SIGIR 2018. Three readers used a manual labelling approach to classify the assertions in the opening parts as summary-worthy or summary-unworthy. It was shown that employing ensembled feature space considerably improved summarization performance when both conventional classification and ROUGE-based analysis were used.

Hiren Kumar Thakkar et al [8] proposed a novel Domain Feature Miner (DOFM) mining algorithm. The summaries generated by DOFM are then subjected to automatic examination using ROUGE. This is a well-known programme for automated assessment of summaries. An error study revealed that 84 percent of the sentences from all DOFM generated summaries were selected by at least one of the three annotators. This highlights the DOFM's resiliency in terms of domain feature retrieval and extractive summarization, as well as its overall performance.

Ángel Hernández-Castañeda et al [9] uses Genetic Algorithm to identify the most effective grouping of words. This model organizes sentences in a text with the assistance of a clustering approach. Summaries generated not only contain matches of unique words, but also give context by matching terms in the text. One-of-a-kind technique for automated summarization is presented in this study. This method can be used to organize sentences in a text according to specific semantic and lexical qualities.It combines a vectorial space formed by a large number of feature generation algorithm(s) with a single summary strategy. It is not necessary to have a prior grasp of the underlying issue in order to generate vectors for this purpose. LDA, Doc2Vec, TF–IDF, and OHE do not need any prior knowledge of classes.

Kaichun Yao et al. [10] developed a extractive document summarizing approach based on Deep Q-Networks (DQN) to capture word salience and redundancy and train a strategy that maximises the Rouge score over gold summaries. The information given by the informative features not only provides informative features to describe the DQN's states but also generates a list of probable DQN actions from the document's words. Our model does not need extractive labels at the sentence level since it is trained directly on human-provided reference summaries. The Rouge measure is used to assess the model's performance on the CNN/Daily, DUC 2002, and DUC 2004 datasets. When applied to non-linguistic corpora, our technique outperforms or is on par with state-of-the-art models in terms of performance. The researchers believe this is the first time DQN has been used for extractive summarization in any scientific setting.

Luca Cagliero et al. [11] mention that annotating scientific articles with textual highlights, it is feasible to provide readers with potentially valuable result-oriented insights that may be used immediately. Unfortunately, the majority of the time, rather than automatically, the annotation process is performed by hand. A further problem is that the highlight information is completely lacking from the vast majority of earlier publications. The solution provided here overcomes the issues noted above by using supervised learning on previously annotated article data.

Jesus M. Sanchez-Gomez et al. [12] using three distinct term-weighting algorithms performed the task of multi-document text summarizing, and the authors found that they were all successful. Different unique similarity metrics that are employed in text-similarity have been taken into consideration by this work. The average and Pearson's coefficient of

variation are two of the computations that were employed in this investigation.

Mohammad Mojrian et al. [13] proposed the MTSQIGA approach, which is a novel multi-document text summarization approach. It is designed to extract salient sentences from a source document collection in order to generate a summary of the information contained in the collection. It is proposed that a modified quantum measurement, as well as a self-adaptive quantum rotation gate, be used in conjunction with a summary generator that is dependent on the quality and length of the summary that is generated. A benchmark dataset from the DUC 2005 and 2007 was used to evaluate the proposed system in terms of ROUGE standard measures.

Akanksha Joshi et al. [14] proposed SummCoder, a method for extracting text summarization from single documents, makes this task much easier. The summary is generated using three metrics: content relevance, novelty, and position relevance. The following are the outcomes: An auto-encoder network is used to determine the relevance of sentence content by exploiting the similarity between embeddings in distributed semantic space, and the novelty metric is derived from this similarity. In this feature, which was created by hand, a dynamic weight calculation function based on the overall length of the document is used to give more weight to the document's first few sentences. It is also possible to create a document summary by ranking the sentences based on a combined final score derived from three different sentence selection metrics. A new summarization benchmark, the Tor Illegal Documents Summarization (TIDSumm) dataset, will benefit law enforcement agencies (LEAs). These summaries were created manually for 100 documents from onion websites in the Tor (The Onion Router) network and are included in the dataset. When compared to other methods, this text summarization approach achieves comparable or better performance for a wide range of ROUGE metrics for the DUC 2002, Blog Summarization, and TIDSumm datasets.

Manh et al. [30] used corpus based measures like LSA and LDA along with K-means to perform the task of summarization. The results of this work are comparatively good as corpus based measures explores different possibilities of evaluation. But the semantic similarity is not explored by Manh et al. [30]. The authors [31], [32] provided the applications of semantic similarity measures for the evaluating verb similarity and sentence with contradictory similarity. The works [31], [32] also discussed the importance of semantic similarity in the current research domains of natural language processing.

Table I provides the summary of the models compared in this article. Table I gives an insight into the models target and whether sentence scoring is performed in the model or not. This section presented various multi-document summarization models based on the different techniques. The models presented in this work are limited to analysing the similarity between the sentence and document title using term weighting schemes. The significance of knowledge based measures is not considered in the works highlighted in this section. We propose a novel knowledge based metric based on information content and path length in the next section.

TABLE I. SUMMARY OF THE RELATED WORK

Ref. No	Model	Target	Sentence Scoring
[5]	Artificial Bee Colony algorithm	Text summarization of multiple documents	Not mentioned
[6]	Semantic Approach	Text summarization of multiple documents	Mentioned
[7]	Candidate sentence selection	Text summarization of multiple documents	Mentioned
[8]	Domain Feature Miner	Text summarization of multiple documents	Not mentioned
[9]	Language-independent Summarization	Keyword Extraction enabled Text summarization	Not mentioned
[10]	Deep reinforcement learning	Text summarization	Not mentioned
[11]	Unsupervised framework	Auto encoder based Text summarization	Not mentioned
[12]	Supervised summarization	Scientific Article Summarization	Not mentioned
[13]	Centroid approach and sentence embeddings	Extractive Text Summarization	Mentioned
[14]	Term-weighting schemes	Text summarization	Mentioned
[15]	Quantum-inspired genetic algorithm:	Text summarization	Mentioned
[16]	Entropy for Extractive Document Summarization	Different measures of text summarization	Not mentioned
[17]	Weighted Word Embedding	Text summarization	mentioned
[18]	Firefly algorithm	Text summarization of multiple documents	Not mentioned
[19]	Fuzzy and evolutionary based model	Extractive Text Summarization	Mentioned
[20]	Summarization of documents	Summarization of image based documents	Not mentioned

The next section gives the proposed model and the significance of knowledge based measures in the sentence similarity evaluation.

III. PROPOSED MODEL

This section presents the proposed work to perform the document summarization. This section also proposes a novel metric using the concepts of semantic similarity to estimate the values of sentence scoring. The first part of this section covers the knowledge-based measures and the second part covers the document summarization aspects.

The measures that produce synonyms and also deal with various word forms are knowledge-based measures. Similarity based on knowledge is determined by the information content or the length between the terms [22]. To determine similarity, knowledge-based methods make use of a well-constructed taxonomy (also known as a lexical database) to infer semantic similarity between concepts.

Information content (IC) is the probability regarding the availability of a concept in the corpus.

$$IC(C_k) = -\log(P(C_k)) \tag{1}$$

Where $P(C_k)$, is probability of the concept Ck in the corpus.

$$P(C) = \frac{f(C)}{N} \tag{2}$$

Where f(C) is the frequency of the concept in the corpus,

N represents the number of words in the corpus.

Path (or path length) between two concepts gives the possible shortest path between the concepts.

The depth (D) is referred as the length of concepts and maximum depth (Dmax) of a concept the length from the concept to root in the taxonomy.

Least common subsumer (LCS) of two concepts in the taxonomy is another concept which is the root of the two concepts.

A. Semantic Similarity Measures

Various semantic similarity measures which are knowledge based are discussed in this subsection. The following are the standard knowledge based semantic measures.

Res Measure [21]: This measure estimates the similarity between the concepts C_i, C_j by considering the information content of the lowest common subsumer.

$$resnik(C_i, C_j) = IC\left(C_{LCS}(C_i, C_j)\right) \tag{3}$$

Jcn [22] Measure: This measure to calculate the similarity between the concepts C_i, C_j proposes the following equation,

$$jcn(C_i, C_j) = IC(C_i) + IC(C_j) - 2 * IC\left(C_{LCS}(C_i, C_j)\right) \tag{4}$$

Lin Measure [23]: This measure is defined as,

$$lin(C_i, C_j) = \frac{2*IC\left(C_{LCS}(C_i, C_j)\right)}{IC(C_i) + IC(C_j)} \tag{5}$$

Lch [24] Measure: This measure considers the maximum depth of the taxonomy and the length of concepts C_i, C_j in the taxonomy.

$$lch(C_i, C_j) = -\log\left(\frac{len(C_i, C_j)}{2*Dmax}\right) \tag{6}$$

Wup [25] (wup) Measure: This measure uses depth of lowest common subsumer of the two concepts C_i, C_j and individual depths of the concepts to estimate the similarity between concepts.

$$wup(C_i, C_j) = \frac{2*D(C_{lcs})}{D(C_i) + D(C_j)} \tag{7}$$

Path Measure [26] (path): This measure calculates the inverse of semantic distance between the concepts C_i, C_j as the similarity between the concepts.

$$pathdistance(C_i, C_j) = length(C_i, C_j) \tag{8}$$

$$path(C_i, C_j) = \frac{1}{1 + pathdistance(C_i, C_j)} \tag{9}$$

Li measure [27] (li): This is a non-linear measure to estimate the similarity of the concepts. This measure uses depth and length between the concepts C_i, C_j to calculate the similarity.

$$li(C_i, C_j) = e^{-\alpha*l} * \frac{e^{\beta*D} - e^{-\beta*D}}{e^{\beta*D} + e^{-\beta*D}} \tag{10}$$

Where α, β are parameters and α= 0.2, β=0.6.

B. Proposed Measure

The metrics indicate that they are all attempting to calculate how much information is shared between them in order to determine how closely two concepts are related. The problem of concepts with the same length and giving the same value even when there is less similarity between the concepts is also not addressed by measures based on the distance between concepts. The issue of equal route length can be solved by adding depth using techniques like wup and li. Greater granularity has the unintended consequence of making the concepts at the top of the hierarchy less detailed. This indicates that the path and depth problems are being addressed using the data. Compared to the route- and depth-based assessments, the information content measurements are more accurate. These measures will give the same similarity score for two concepts that have the same LCS, regardless of how differently their contents are expressed. For this problem, the information content serves as a guiding weight, and the measure may be represented as follows:

$$Hybrid\ measure(C_i, C_j) = \frac{1}{1 + path(C_i, C_j) \times k^{IC(2* C_{LCS}(C_i, C_j))/(IC(C_i) + IC(C_j))}} \tag{11}$$

The suggested metric has various weights for the path length, which eliminates the issue of ideas having the same path length and hence having the same LCS difficulties. Conceptual similarity is estimated by taking into consideration the semantic distance between ideas and their respective information content as well as the information content of each concept measured separately (LCS).

The sentence scoring is calculated by deriving a sentence feature vector. The sentence feature vector is calculated by using the NLTK and proposed semantic similarity measure.

C. Proposed Extractive Multi-Document Summarization

In this section, we discuss our proposed system. Fig. 1 gives the architecture of the proposed model. The input to the model is a set of documents. The documents are taken from reliable datasets.

The first phase in the model is preprocessing of data. In the preprocessing, sentence segmentation is performed initially. Later the sentences are tokenized, and each sentence is represented as a set of tokens at this step. The parts-of-speech tagging relative to the words in the sentence is also preserved. The most irrelevant words from the sentence are removed and the remaining words are stemmed.
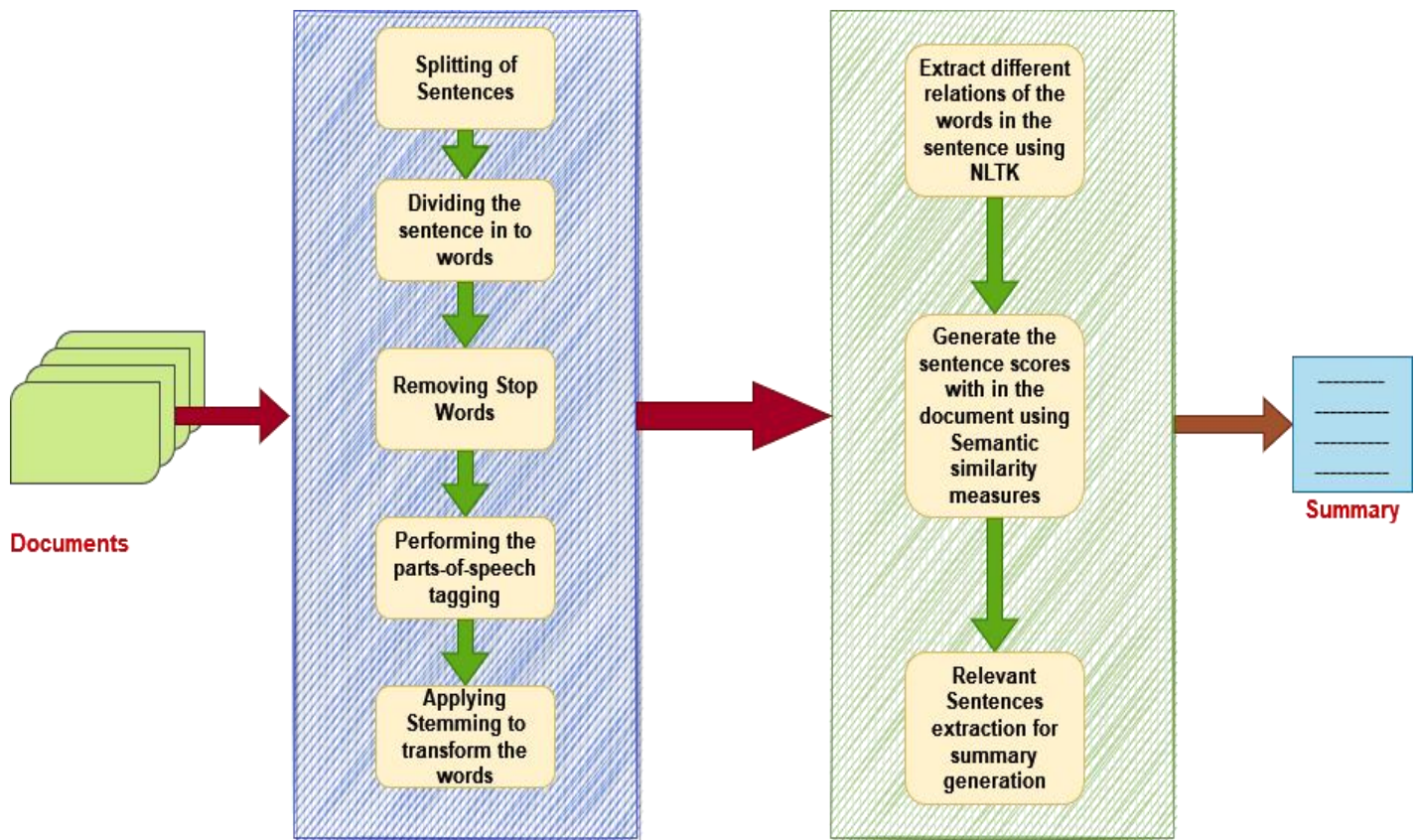
Fig. 1. Architecture of the Proposed Model.

The next stage in the proposed model is to extract multiple senses of each word preserved in the sentence. The relations are extracted from the NLTK package. The next step in the second stage is to generate the sentence scores. The sentence scores are used to generate the summary of the documents. Later the performance metrics are used in the evaluation.

Preprocessing is accomplished by the use of a pipeline that is often used for multi-document summarizing jobs, in which a cluster of documents is represented by a collection of phrases. In other words, we're discussing the presence of a cluster D containing m documents D = [d1, d2,..., dm].

The initial step was to decompose each document di in the cluster D into individual phrases, which we performed with the help of the free and open-source software package spaCy for Advanced Natural Language Processing. The next phase uses the Natural Language Toolkit (NLTK) and regular expressions to clean up these phrases by converting all words to lower case and deleting special characters, unnecessary whitespace, HTML elements, URLs, and email addresses from the source code.

D. Semantic Relationship between Words

We employ semantic similarity metrics and WordNet to capture the semantic links between words. We begin by assessing the word's resemblance to the document title. The highest degree of similarity that a word achieves is referred to as the word score. The TF-IDF, a term weighting approach, is used to weight these word scores.

We use multiplication to integrate TF-IDF and word scores in our current work. To create a sentence vector, we integrate all of the sentence's word weighed scores. The TF-IDF assists in mapping the phrase to a distributed semantic vector, with the exception that the most frequently occurring words have a smaller influence on the outcome. Finally, we acquire a vector representing all sentences in the corpus; this vector is referred to as the average sentence vector.

E. Extracting Different Features

The linguistic features of the sentence are also extracted as important features to calculate the sentence score. Representation of the calculations of the sentence vector using different features is mentioned in Fig. 2.

1) Noun and verb phrase: The noun or verb phrases in a sentence are essential and given more weight in a sentence. Each sentence's noun and verb phrase weight is computed as follows:

$$NV_{Phrases} = \frac{No.(Verbs, Nouns)}{Sentence\ Length}$$

2) Sentence position: In general, the sentences at the beginning and end are more informative.

3) Sentence length: Sentences with large length are more significant in the document.

Proposed Algorithm

Input: Document
Output: Summary

Step 1: Preprocessing of document d,

- d <--- Segment (d) // Sentence Segmentation
- S <--- Tokenize (d) // Tokenization of each sentence
- S <-- Punct_Rem (S) // Punctuation Removal
- S <-- Stopword_removal (S) // Stop word Removal
- S <-- Stem (S) // Stemming

Step 2: Sentence Vector Generation

for each sentence

- WS_i <-- Word_Score(S_i) // Word score is the maximum similarity for each word in relevance with the document title.
- WWS_i <-- TF_IDF * WS_i // Weighted word scores
- SV_i <-- WWS_i

Step 3: Different Feature Weights

NV <-- count (nouns, verbs)
SL <-- Length (S_i)

Step 4: Weight of each sentence

for each sentence

SW_i <-- Weight(NV, SL, SV_i)

Step 5: Semantic importance of each sentence
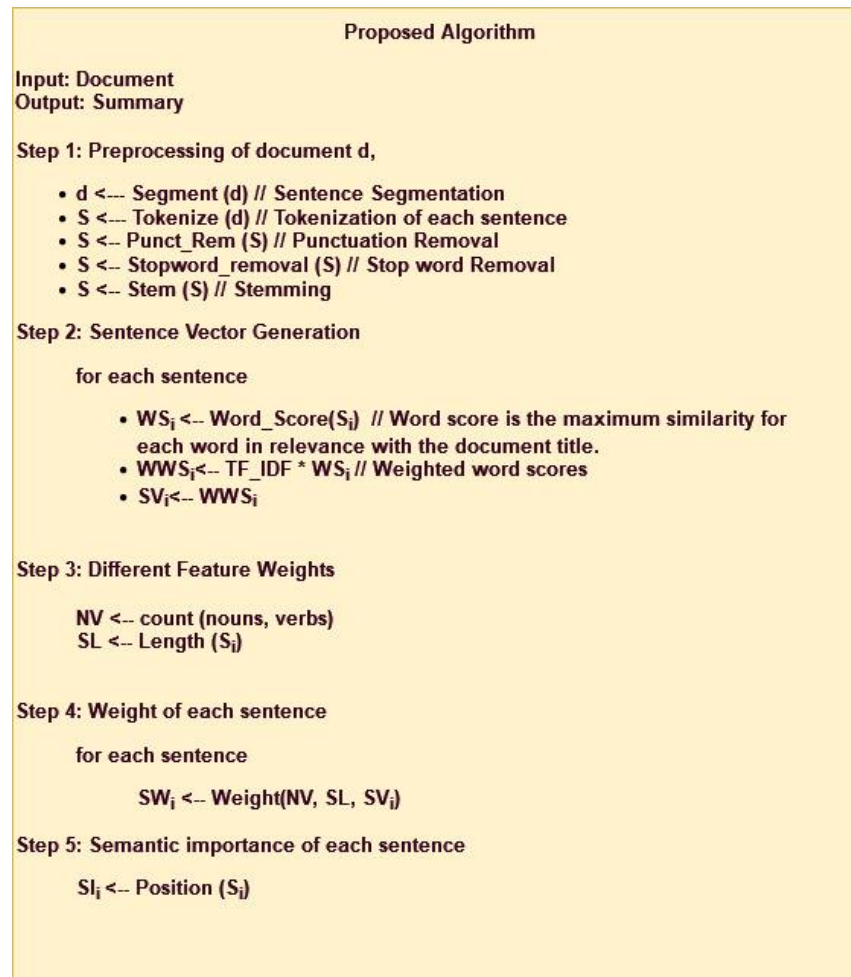
SI_i <-- Position (S_i)

Fig. 2. Proposed Algorithm.

Based on the features and sentence vectors the sentence scores are generated and the sentences with higher scores are considered as more relevant sentences and these sentences are combined together to generate the summary of sentences. The different steps involved in calculating the sentence scores are mentioned in the proposed algorithm. The different features used for scoring the sentence are discussed above.

This section covers the overall description of the proposed model and the results regarding these models are presented in the next section.

IV. RESULTS AND DISCUSSION

This section covers the experimentation of the proposed similarity measure on word pair similarity dataset and the proposed model to perform multi-document summarization of data. The first part of the results is with respect to the semantic similarity on different word pair datasets.

A. Metrics Used

Pearson correlation: This correlation is used to evaluate the performance of various semantic similarity measures.

Spearman Correlation: This is also another well-known correlation that is used to evaluate the performance of various semantic similarity measures.

ROUGE Score: Many researchers and practitioners use this metric to assess how well multi-document summarizing algorithms function.

B. Datasets

- RG dataset [28]: This dataset, which includes 65 noun pairs, is used to assess word similarity tasks.

- MC dataset [29]: This dataset, which includes 30 noun pairs, is used to assess word similarity tasks.

- DUC 2007 dataset: The dataset consists of 45 separate subjects that are each covered by 45 unique documents, all of which cover all 45 categories.

C. Tools used for Implementation

- NLTK

- Spacy

- ROUGE

- SCIKIT learn

- Anaconda

The results of several semantic similarity tests performed on the RG dataset and the MC dataset are shown in Tables II and III, respectively. The findings show that combining the length between the concepts with the information content leads in greater correlation values. The proposed hybrid measure is able to achieve better results when compared with all the existing models.

As a baseline for our current research, we utilise the primary task dataset from the Document Understanding Conference (DUC 2007). Automated text summarization assessment is carried out by NIST using the DUC 2007 dataset. The DUC 2007 dataset is made up of news stories from a variety of publications. The dataset contains 45 separate subjects and 45 individual texts, each of which discusses all 45 themes.

Fig. 3 is an example of how summarization works. The proposed model after sentence scoring extracts the relevant sentences according to the document in the shown example. The results of the summarization are with respect to a compression rate of 15%. It can be observed from the figure that the relevant sentences are extracted according to the given data.

TABLE II. RESULTS OF CORRELATION ON RG DATASET

Measure	Spearman	Pearson
path	0.78	0.78
li	0.79	0.86
lin	0.78	0.86
res	0.78	0.84
lch	0.78	0.84
wup	0.76	0.79
jcn	0.78	0.72
Proposed	**0.80**	**0.86**

TABLE III. RESULTS OF CORRELATION ON MC DATASET

Measure	Spearman	Pearson
path	0.78	0.78
li	0.79	0.86
lin	0.78	0.86
res	0.78	0.84
lch	0.78	0.84
wup	0.76	0.79
jcn	0.78	0.72
Proposed	**0.80**	**0.86**

Because of the massive amount of textual information accessible today, automated extraction text summarization is one of the most extensively used ways to organise the information. The summarization mechanisms helps to extract the important topics of data from a given set of documents. One of the summarization techniques named extractive summarization selects the most relevant sentences from the text to provide a representative summary. The main purpose of extractive multi-document text summarization systems is to reduce the amount of textual information in a document collection by focusing on the most important topics and avoiding unnecessary portions. In the previous research, there are several methods such as term-weighting schemes and similarity metrics used for constructing an automated summary system. There are few studies that look at the performance of combining various Semantic similarity and word weighting techniques in automatic text summarization. We evaluated numerous semantic similarity metrics in extractive multi-document text summarization in this research. In this paper, we explored various semantic similarity measures in the extractive multi-document text summarization. Experiments have been performed with Document Understanding Conferences (DUC) datasets, and the model performance has been assessed with eight Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metrics. Even more, the combination formed by different semantic similarity measures obtained the highest results in comparison with the other models.

The main purpose of text summarization systems is to reduce the amount of textual information in a document collection by focusing on the most important topics and avoiding unnecessary portions. One of the summarization techniques named extractive summarization selects the most relevant sentences from the text to provide a representative summary. There are few studies that look at the performance of combining various Semantic similarity and word weighting techniques in automatic text-editing systems.

Fig. 3. An Example of Extractive Summarization of the Proposed Model.

Because each subject has 45 subtopics, we may construct a summary for each of them in our suggested approach. For each of the comparative models and our recommended strategy, we've compiled summaries. Summaries of various sizes were created using different compression ratios of 5 percent, 15%, 25%, and 50% of the original material. The ROUGE-N score metric assesses the quality of the summaries that were created for the purposes of this section.

When it comes to automated summaries, ROUGE is often regarded the gold standard. Rouge contrasts the summaries produced by machines with the summaries created manually (reference summaries). ROUGE-1, ROUGE-2, and ROUGE-L summaries are evaluated at various levels of granularity, giving findings in terms of Precision (P), Recall (R), and F-score (F). For evaluation, ROUGE-1 is used and the results articulated in the following tables are for ROUGE-1. The results of the proposed model on DUC 2007 dataset are articulated in the Tables IV, V and VI.

TABLE IV.   AVERAGE F-SCORE ROUGE SCORE VALUES AT COMPRESSION 5% , 15%, 25% RATES

Measure	ROUGE Score (5%)	ROUGE Score (15%)	ROUGE Score (25%)
Tf-idf	0.22	0.23	0.22
Tf-isf	0.22	0.23	0.22
Rtf-sisf	0.23	0.22	0.23
Okapi BM25	0.22	0.24	0.22
Resnik-tf-idf	0.24	0.23	0.24
Resnik-tf-isf	0.24	0.25	0.24
Resnik-Rtf-sisf	0.24	0.25	0.26
Hybrid-Tf-idf	**0.26**	**0.26**	**0.27**

TABLE V.   AVERAGE PRECISION ROUGE SCORE VALUES AT COMPRESSION 5%, 15%, 25% RATES.

Measure	ROUGE Score (5%)	ROUGE Score (15%)	ROUGE Score (25%)
Tf-idf	0.23	0.23	0.22
Tf-isf	0.23	0.23	0.22
Rtf-sisf	0.24	0.22	0.23
Okapi BM25	0.23	0.24	0.22
Resnik-tf-idf	0.25	0.23	0.24
Resnik-tf-isf	0.24	0.25	0.24
Resnik-Rtf-sisf	0.26	0.25	0.26
Hybrid-Tf-idf	**0.28**	**0.28**	**0.28**

TABLE VI.   AVERAGE RECALL ROUGE SCORE VALUES AT COMPRESSION 5% , 15%, 25% RATES

Measure	ROUGE Score (5%)	ROUGE Score (15%)	ROUGE Score (25%)
Tf-idf	0.24	0.25	0.24
Tf-isf	0.24	0.25	0.24
Rtf-sisf	0.25	0.24	0.25
Okapi BM25	0.24	0.26	0.24
Resnik-tf-idf	0.26	0.25	0.26
Resnik-tf-isf	0.26	0.28	0.26
Resnik-Rtf-sisf	0.26	0.28	0.29
Hybrid-Tf-idf	**0.29**	**0.29**	**0.30**

When compared to the literature, our experimental data has shown that our suggested strategy outperforms the state-of-the-art methodologies, which we feel is important. According to the findings, Table IV further demonstrate that the average Recall values across a variety of variables improve as a consequence of increasing the length of the summary, as can be seen in the tables. Because our model's recall is lower in certain places than it is in others, it is possible that this is due to either a shorter summary or the removal of statistically important characteristics from the model's development process throughout its development. Following an increase in the compression rate from 5 percent to 25 percent, the macro-average F-score values decline somewhat as a consequence of a reduction in the overall accuracy score of the different metrics when the compression rate is raised, according to the study's findings.

However, when comparing the Macro-Averaged F-score values at 22 percent and 23 percent compression rates to the comparative models, as shown in Tables IV, V and VI, the difference is not statistically significant; the difference between the two models is not statistically significant. This demonstrates that the approach given is competitively efficient when compared to the current state of the art. In Table VI, it is shown that, when constructing an average length summary at a 25 percent compression rate, the suggested technique may result in a summary that is more informative than comparison models in certain cases.

This section presented the results of the proposed model and proposed hybrid semantic similarity on word pair similarity and DUC 2007 datasets. The presented results show the efficiency of the model.

V.   CONCLUSION

In light of the vast amount of textual information that is now available, automated extraction text summarization is one of the most widely used methods of organising the data available. Summary techniques make it possible to extract the most important information from a large number of texts in a short amount of time and with minimal effort. When summarizing a text, an extractive summarization method is used that selects the most relevant phrases from the text and presents them in a way that is accurate representation of the text in its entirety. Information extraction systems that extract

information from a large number of documents, such as text summarizing systems, have as their primary objective the reduction of textual information in a document collection. Achieving this is accomplished by concentrating on the most important themes and eliminating any unnecessary information. When it came to developing an automated summary system, the previous study discovered that a variety of strategies, including term-weighting schemes and similarity metrics, were used in the process of development. Currently, there is only a small body of research that examines how different Semantic Similarity and word weighting algorithms perform when used in conjunction with one another in the field of automated text summarization. This study looked at a number of different semantic similarity metrics in the context of extractive multi-document text summary, and we discovered that they were all fairly accurate in terms of similarity. This research looked into different semantic similarity metrics that could be used in extractive multi-document text summarization, and the results were published. Various ROUGE criteria were used to evaluate the model's performance in this study, which was carried out using DUC dataset. When the results of the various semantic similarity metrics were combined, the resulting model produced the most favourable results when compared to the other models in this study.

REFERENCES

[1] A. K. Srivastava, D. Pandey, and A. Agarwal, "Extractive multi-document text summarization using dolphin swarm optimization approach," Multimedia Tools and Applications, vol. 80(7), pp. 11273-11290, 2021.

[2] W. S. El-Kassas, C. Salama, A. A. Rafea, and H. K. Mohamed, "Automatic text summarization: A comprehensive survey," Expert Systems with Applications, vol. 165, pp. 113679, 2021.

[3] L. Dong, M. N. Satpute, W. Wu, and D. Z. Du, "Two-phase multidocument summarization through content-attention-based subtopic detection," IEEE Transactions on Computational Social Systems, vol. 8(6), pp. 1379-1392, 2021.

[4] J.M. Sanchez-Gomez, M. A. Vega-Rodríguez, and C. J. Perez, "Parallelizing a multi-objective optimization approach for extractive multi-document text summarization," Journal of Parallel and Distributed Computing, vol. 134, pp. 166-179, 2019.

[5] J.M. Sanchez-Gomez, M. A. Vega-Rodríguez, and C. J. Perez, "A decomposition-based multi-objective optimization approach for extractive multi-document text summarization," Applied Soft Computing, vol. 91, pp. 106231, 2020.

[6] M. Bidoki, M. R. Moosavi, and M. Fakhrahmad, "A semantic approach to extractive multi-document summarization: Applying sentence expansion for tuning of conceptual densities," Information Processing & Management, vol. 57(6), pp. 102341, 2020.

[7] B. Mutlu, E. A. Sezer, and M. A. Akcayol, "Candidate sentence selection for extractive text summarization," Information Processing & Management, vol. 57(6), pp. 102359, 2020.

[8] H. K. Thakkar, P. K. Sahoo, and P. Mohanty, "DOFM: Domain Feature Miner for robust extractive summarization," Information Processing & Management, vol. 58(3), pp. 102474, 2021.

[9] A. Hernández-Castañeda, R. A. García-Hernández, Y. Ledeneva, and C. E. Millán-Hernández, "Language-independent extractive automatic text summarization based on automatic keyword extraction," Computer Speech & Language, vol. 71, pp. 101267, 2022.

[10] K. Yao, L. Zhang, T. Luo, and Y. Wu, "Deep reinforcement learning for extractive document summarization," Neurocomputing, vol. 284, pp. 52-62, 2018.

[11] A. Joshi, E. Fidalgo, E. Alegre, and L. Fernández-Robles, "SummCoder: An unsupervised framework for extractive text summarization based on

[12] L. Cagliero, and M. La Quatra, "Extracting highlights of scientific articles: A supervised summarization approach," Expert Systems with Applications, vol. 160, pp. 113659, 2020.

[13] S. Lamsiyah, A. El Mahdaouy, B. Espinasse, and S. E. A. Ouatik, "An unsupervised method for extractive multi-document summarization based on centroid approach and sentence embeddings," Expert Systems with Applications, vol. 167, pp.114152, 2021.

[14] J.M. Sanchez-Gomez, M. A. Vega-Rodríguez, and C. J. Perez, "The impact of term-weighting schemes and similarity measures on extractive multi-document text summarization," Expert Systems with Applications, vol. 169, pp. 114510, 2021.

[15] M. Mojrian, and S. A. Mirroshandel, "A novel extractive multi-document text summarization system using quantum-inspired genetic algorithm: MTSQIGA," Expert systems with applications, vol. 171, pp. 114555, 2021.

[16] A. Khurana, and V. Bhatnagar, "Investigating Entropy for Extractive Document Summarization," Expert Systems with Applications, vol. 187, pp. 115820, 2022.

[17] R. Rani, and D. K. Lobiyal, "A weighted word embedding based approach for extractive text summarization," Expert Systems with Applications, vol. 186, pp. 115867, 2021.

[18] M. Tomer, and M. Kumar, "Multi-document extractive text summarization based on firefly algorithm," Journal of King Saud University-Computer and Information Sciences, vol. 34(8), pp.6057-6065, 2021.

[19] P. Verma, A. Verma, and S. Pal, "An approach for extractive text summarization using fuzzy evolutionary and clustering algorithms," Applied Soft Computing, vol. 120, p.108670, 2022.

[20] J. Chen, and H. Zhuge, "Extractive summarization of documents with images based on multi-modal RNN," Future Generation Computer Systems, vol. 99, pp. 186-196, 2019.

[21] Resnik, "Using information content to evaluate semantic similarity in a taxonomy," arXiv preprint cmp-lg/9511007, 1995.

[22] Harispe, Ranwez, Janaqi, and Montmain, "Semantic similarity from natural language and ontology analysis," Synthesis Lectures on Human Language Technologies, vol. 8(1), pp. 1-254, 2015.

[23] Jiang, and David W Conrath, "Semantic similarity based on corpus statistics and lexical taxonomy," arXiv preprint cmp-lg/9709008, 1997.

[24] Lin, and Dekang, "An information-theoretic definition of similarity," Proceedings of ICML, vol. 98, 1998.

[25] Leacock, Claudia, and Martin Chodorow, "Combining local context and WordNet similarity for word sense identification," WordNet: An electronic lexical database, vol. 49(2), pp. 265-283, 1998.

[26] Wu, Zhibiao, and Martha Palmer, "Verbs semantics and lexical selection," Proc. Asso. Comp. Ling, 1994.

[27] Rada, Roy, et al, "Development and application of a metric on semantic nets," IEEE trans. on syst., man, and cyber. vol. 19.1, pp.17-30, 1989.

[28] Li, Bandar and McLean, "An approach for measuring semantic similarity between words using multiple information sources," IEEE Trans.on know. and data eng., vol. 15(4), pp. 871-882, 2003.

[29] Rubenstein, Herbert, and John B. Goodenough, "Contextual correlates of synonymy," Commun. ACM, vol. 8(10), pp. 627-633, 1965.

[30] Manh, Hai Cao, Huong Le Thanh, and Tuan Luu Minh, "Extractive Multi-document Summarization using K-means, Centroid-based Method, MMR, and Sentence Position," Proceedings of the Tenth International Symposium on Information and Communication Technology, pp. 29-35, 2019.

[31] M. Krishna Siva Prasad, and Poonam Sharma, "Similarity of Sentences With Contradiction Using Semantic Similarity Measures," The Computer Journal 65, no. 3 (2022): 701-717.

[32] M. Krishna Siva Prasad, and Poonam Sharma, "Exploring intrinsic information content models for addressing the issues of traditional semantic measures to evaluate verb similarity." Computer Speech & Language 71 (2022): 101280.