Diverse Approaches in News HeadLine Generation

***Abstract*— Due to extent of information, it is practically impossible to process all the data by any single entity, hence move to the headline generation process. Further, consumers might not be interested in reading long text, which makes them to skip the important portions of the text. This aspect increases the demand of automatic text headline generation. Researchers are mostly concentrating on these, however, the domain- specific multiple data headline generation with topic modeling is limited. Aiming in this direction, a novel text headline generation approach with domain specific topic modeling referred as OBSum is introduced in this work. This approach encapsulates five major stages: "Pre-processing, topic identification via tokenization, mapping text to contextualized embeddings, sentence clustering, and sentence selection". The collected raw data is pre-processed and the domain –specific topic are identified from it using the token influence score based tokenization approach. The OBSum is utilized in this research work to map the sentences to contextualized embeddings. In OBSum framework, the weight and biases of TRANFORM will be fine-tuned. At the end, the summary is generated by selecting informative sentences from all the clusters. The proposed multi-document text headline generation is compared over the existing models in terms of certain performance measures like ROUGE 1, ROUGE 2 as well.**

***Keywords—Text Headline generation; Topic Modelling;Token Influence Score based Topic Identification;OBSum;* SAPSO.**

Nomenclature

Abbreviation	Description
GATS	Generating Abstractive Text Summaries
ATS	Abstractive Text Headline generation
TRANFORM	Bidirectional Encoder Representations From Transformers
Bi-LSTM	Bidirectional Long Short-Term Memory
CNNs	Convolutional Neural Networks
NLP	Natural Language Processing
PSO	Particle Swarm Optimization
RNN	Recurrent Neural Networks
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
SAPSO	Self Adaptive Particle Swarm Optimization
TF-IDF	Frequency-Inverse Document Frequency
VAE	Variational Auto-Encoder
DNN	Deep Neural Networks
PGM	Probabilistic Generative Model
GETS	Generic Extractive Text Summarization
TIDSumm	Tor Illegal Documents Summarization
ASTR	Abstractive Review Summarization With Topic Modeling And Reinforcement Deep Learning

I. INTRODUCTION

A pivotal role is being played by the automated text processing tools for effectual acquaintance of knowledge from enormous sources of documented information particularly in health care and life science like the "scientific publications, electronic health records or clinical guidelines". In this era, the internet is being a knowledge source for humans and the textual data generated by it is massive [3] [9] [10] [11]. This tends to be more crucial and challenging since the consumers won't be interested in reading long text. This brings the necessity of novel tool to characterize the content in a concise form called summary. An important branch of NLP [22] [23] [24] [25] is the Automatic text summarization that intents to describe the extended text documents in a compact manner, such that the end users can quickly read and understand the information. The text summarization techniques are grouped as "extractive and abstractive text summarization" [3] [12] [14] [13] [15] [16]. In the Extractive text summarization, the concatenation of the most relevant sentences is done to generate the final summary of the document. Conversely in abstractive text summarization, main contents are paraphrased using NLP generation techniques to generate the final summary. Hence, the overall idea of the contents in the document is provided by the generic summarization [3].

The traditional models in the literature concern the extractive text summarization that is based on the human-engineered features like "combination of statistical and linguistic features such as term frequency, sentence length and position, or cue and stigma words". Further, various techniques like the meta-heuristic optimization based approaches, graph-based approaches, greedy approaches as well have been proposed for summary generation from the selected sentences [3] [17] [18] [19] [20] [21].

In more recent works done in NLP tasks like the "question-answering, sentiment analysis, natural language understanding, text classification and language translation", the deep learning based methods are have attained impressive accuracies[3]. Recently, the introduction of the supervised deep learning approaches is the biggest challenge for extractive text summarization for training the networks. The deployment of the auto-encoders for generic for Arabic text summarization utilized the TF-IDF vectors [3] [26]. But, here the convergence and the accuracy of the text summarization are lower. Therefore, the optimization algorithms [27] [28] [29] [30] can be deployed as a solution for text summarization.

The major contribution of this research work is:

- ✓ A novel token influence score based tokenization approach is introduced that states the domain specific summarization.
- ✓ OBSum is introduced for mapping the sentences to contextualized embeddings. Here, the weight and biases of TRANFORM is be fine-tuned by a new SAPSO algorithm.

The rest of the paper is organized as: Section II provides a review on the most recent works in literature. Section III portrays about the proposed multi-document text summarization with domain specific topic modelling.

The pre-processing and topic identification via token influence score are depicted in Section IV. Further, text mapping to contextualized embeddings, sentence clustering and sentence selection are addressed in Section V. The results acquired are discussed in a comprehensive manner in Section VI.

II. LITERATURE REVIEW

A. Related works

In 2019, Moradi *et al.* [1] have generated a novel summarization method on the basis of the contextualized embeddings by TRANFORM model. The TRANFORM-based biomedical summarizer encapsulates four major phases like, "pre-processing, mapping text to contextualized embeddings, sentence clustering, and sentence selection". From the input documents, the informative sentences and the most relevant sentences were identified by means of combining the clustering method and the different versions of TRANFORM. The proposed summarizer was evaluated against several methods in literature using the ROUGE toolkit.

In 2020, Zhao *et al.* [2] have developed SummPip for "multi-document summarization". In the proposed method, the original documents were transformed into a sentence graph by means of considering both the deep and linguistic representation. The authors have acquired the multiple clusters of sentences by deploying the spectral clustering and the final summary was generated by compressing each clusters. The resultant of the proposed model had exhibited its enhancement over existing models in terms of consistency and reliability with "Multi-News and DUC-2004 datasets".

In 2019, Joshi *et al*. [3] have projected SummCoder for GETS of single documents and here the authors have generated the summary with respect to three selected sentences metrics, namely relevance of the sentence position, novelty of sentence and relevance of the sentence content. In addition with the deep auto-encoder network and in a distributed semantic space, the similarity among sentences in the form of embeddings was exploited by deriving the novelty metrics. The three sentence selection metrics were fused and on the basis of their final score, the document summary was generated. A new summarization benchmark, TIDSumm dataset was generated to evaluate the proposed work.

In 2019, Anand and Wagh [4] have proposed ASTR for supervised DNN, reinforcement learning, and unsupervised PGM based deep learning for representing the aspect/sentiment-aware review. The proposed approach was a multi-task learning system that had merged two major objectives: "domain classification (auxiliary task) and abstractive review summarization (primary task)". In addition, the authors have proposed a weakly supervised LDA model with the aim of gathering knowledge on sentiment lexicon and domain-specific aspect representations, which were fed as input to the neural hidden states for aspect/sentiment-aware review representations.

In 2018, Qasem A. Al-Radaideh and Dareen Q.Bataineh [5] have developed a hybrid single document approach (ASDKGA) which incorporates domain knowledge, statistical features and genetic algorithm to extract important points. Further, the approach is tested on KALIMAT and EASC. The proposed approaches are compared with ROUGE.

In 2019, Song *et al*. [6] have introduced ATS for summary sentence creation from different source sentences and to preserve the shorter representation with no loss in information. The projected LSTM-CNN based ATS framework (ATSDL) had the potential of constructing new sentence by means of exploiting the semantic phrases that was more fine-grained fragments than sentences. Finally, the proposed work has shown reliable results from CNN.

In 2018, Alami *et al*. [7]have constructed a new approach for Traditional ATS using VAE model to learn the feature space from the "high-dimensional input data". On the basis of the latent representation generated from VAE, the authors have ranked the sentences. The query-based and graph-based approaches were utilized for investigating the impact of the presented work.

In 2020, Tomer and Kumar [8]have introduced a novel hybrid approach for GATS. The proposed work was the amalgamation of FLS for extractive sentence selection with Bi-LSTM for abstractive summary production. Moreover, the network weights were updated using the attention mechanism and Adam optimizer. They have explored the most relevant sentences from the document fuzzy measures and inference. An abstractive summary was produced for the significant sentences by feeding the relevant sentences as input to Bi-LSTM.

B. Review

Table I shows the features and challenges of existing text summarization and topic modeling approaches. The TRANFORM[1] improves the performance of biomedical text summarization and here there is a need for decreasing the over fitting or lack of generalization. The SummPip in [2] produces consistent and complete summaries. The fluency, consistency, convergence and redundancy can be improved. SummCoder [3], the processing speed is higher. But, Computational cost is higher. Further, ASTR in [4] can generate better sentiment-aware summarization. The Computational cost can be reduced. The ASDKGA approach in [5] can reduces perusing time. On the other hand, the loss of synonym detection need to be considered. LSTM-CNN in [6] improves the effect of phrase acquisition. Apart from this, the proposed approach hold good for smaller datasets. The VAE in [7] learns data from a richer latent space with reduced dimensions. For better accuracy, the noise in the input data needs to be lessened. Further, Bi-LSTM in [8] can pick out specific elements selectively from that sequence. But, time of training is higher. Therefore, the need for automatic text summarization is an urgent requirement.

TABLE I. FEATURES AND CHALLENGES OF EXISTING WORKS

Author [Citation]	Adapted Methodology	Features	Challenges
Moradi *et al.* [1]	TRANFORM	• Low computational complexity • Shown high correlations with informativeness scores	• Need to investigate the impact of model size • Need to decrease the overfitting or lack of generalization
Zhao *et al.* [2]	SummPip	• Produces consistent and complete summaries • Produces high-quality summaries	• The fluency, consistency, convergence and redundancy can be improved. • Need ranging approach for better reliability
Joshi *et al.* [3]	SummCoder	• Lower errors • Higher processing speed	• Higher Computational cost
Anand and Wagh [4]	ASTR	• Can generate better sentiment-aware summarization	• The computational cost can be reduced.
Dareen and Quasem[5]	ASDKGA	• Reduces perusing time • Determination procedure simple	• Lack of synonym detection • Does not able to summarize multiple document
Song *et al.* [6]	LSTM-CNN	• The phrase acquisition effect is reduced. • The phrase redundancy is reduced	• Time consuming • Requires higher training applicable for smaller datasets
Alami *et al.* [7]	VAE	• Lower training time • Reduces the dimensionality of the matrix	• Content overlap need to be reduced • The noise in the input data need to be lessened
Tomer and Kumar [8]	Bi-LSTM	• Exhibits flexibility • Easy to implement and integrate • Summary is generated with larger dataset	• Time of training is higher

III. PROPOSED TEXT HEADLINE GENERATION WITH DOMAIN SPECIFIC TOPIC MODELLIG

In this research work, a novel text modelling approach referred as OBSum is designed for multi-document data text headline generation by means of following five major stages: "Pre-processing, topic identification via tokenization, mapping text to contextualized embeddings, sentence clustering, and sentence selection". An overall architecture of the proposed topic modelling approach is illustrated in Fig.1. Initially the collected raw data D_{input} are pre-processed $(D_{pre-process})$ and the abstract "topics" that occurs are found using a token influence score based tokenization approach (D_{topic}). The fixed number of topics will be known in advance and based on the domain specifications, the topic representation for each document and the words associated with each topic will be learned by considering them as tokens. Then, under a specific topic, the sentence within the topic is mapped to an "n-dimensional vector of real numbers". The OBSum is designed in this research work to map the "sentences to contextualized embeddings". In OBSum framework, the weight and biases of standard TRANFORM is fine-tuned by proposed SAPSO model. The mapped sentence is denoted as D_{map} .The Contextualized embeddings generated for each of the tokens and sentences are captured in the way in which they appear. These generated contextualized embeddings are subjected to sentence clustering by the summarizer to group the similar sentences together. Then, the clustered similar context from the clustering step covers a wide range of important contents with the sentence selection strategy. At the end, the final summary is generated by selecting $D_{summary}$ informative sentences from all the clusters. A detailed description of all these steps is discussed in the upcoming section.

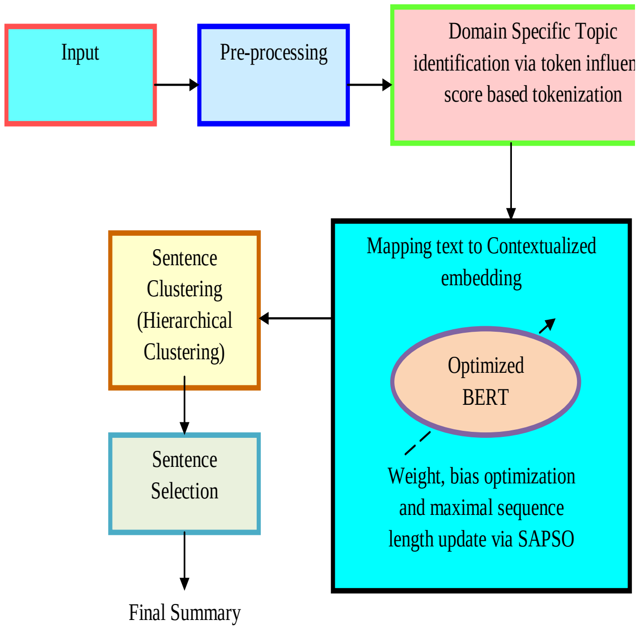


Fig. 1. Overall Architecture of Text Headline generation Model

IV. Pre-Processing and Topic Identification via Token Influence Score

The text headline generation with domain specific topic modelling is introduced in this work with specific steps followed.

A. Pre-Processing

This is the fundamental step and here the unnecessary parts are discarded from D_{input}. In case of BBC news data, all numbers, signs, symbols, non English letters, stop words are removed, stemming is performed, and then all English letters are converted to lower case. In case of BBC sports data, the collected data are converted to lower cased to remove the ambiguity of words. Lower cased data are subjected to pre-processing via Stop word removal and stemming. In case of DUC 2002 data, normalization is preformed in the pre-processing stage to discard stop words, spaces and unwanted characters. In opinosis, the pre-processing is accomplished via sentence embedding technique.

The pre-processing stage further proceeds by means of splitting D_{input} into individual sentence, from which the domain specific topics are determined via extracting the tokens. Then, with the Natural Language ToolKit (NLTK), the main text of D_{input} is split into set of sentences S by the summarizer. Thereby, the topic T in each of the sentences S is the set of tokens, which are extracted by a novel token influence score evaluation.

B. Token Influence Score based Tokenization: A Topic identification Approach

The topic modeling concept encapsulates entities like words, documents, and corpora. In a document, the "Word" is referred as a fundamental unit of discrete data and the document is the organization of N words. In addition, the collection of M documents is said to be corpus and in plural form it is defined as corpora. The topic is nothing but the distribution of the number of fixed vocabulary. In the corpus, each of the documents has its own proportions of discussed topics as per the contained words. The topic modelling can be defined as the simple way of analyzing large volume of texts without labels. "Topic" that has a "group of words that often occur together". The traditional topic modelling approaches are time consuming and are not domain specific in most of the time. They may fail to analyze the data more accurately, while large datasets are taken into consideration. Therefore, in this research work a novel tokenization based topic identification model is introduced, which makes the text headline generation more precise.

The procedure for token specified topic determination is shown below:

(a) The token T (topic) distribution α is considered.

(b) Each sentence S in the document d is assigned to one of the token T

(c) For each S in d

- Compute token influence score I_1, I_2, I_3 as shown below:

Let's consider 5 sentences S_1, S_2, S_3, S_4, S_5 with corresponding topics, where each topic is considered as a token (T). Let $\{T = T_1, T_2, T_3, T_4, T_5\}$ be the token set for the entire sentence i.e. $t_i : i = 1,2,3,..|T|$ and $s_j : j = 1,2,3,..|S|$. In addition, the subset of token s_j is $s_j \subseteq T$. The sentences and tokens in $D_{pre-process}$ taken for illustration are shown in Table 1.

TABLE II. Sentence and Tokens of Input Data: An Illustration

Sentence	Tokens				
S_1	T_1	T_2	T_3		
S_2	T_1	T_2	T_3	T_4	T_5
S_3	T_1	T_2	T_3	T_4	
S_4	T_1	T_2	T_3	T_4	
S_5	T_1	T_2	T_3	T_4	T_5

Step 2: For each of the sentence, the token influence score is computed based on the frequency of the occurrence of the tokens.

For illustrating the computation of the token influence score, let's consider for $S_1 = \{t_1, t_2, t_3\} \subseteq \{T\}$.

I_1 **Calculation:** The Token influence score for I_1 can be computed as per Eq. (1), where, $F_{tk}(k)$ is the frequency of token k_t in $s_j : j = 1,2,3,..|S|$.

$$I_{1j} = \frac{1}{|S_j|} \sum_{t_k \in S_j} \frac{1}{F_{tk}(k)} \tag{1}$$

For illustration, while computing the occurrence of a topic (token) in sentence 1, the computation can be made as per Eq. (2). Here, T_{k1}, T_{k2}, T_{k3} appears in all 5 sentences, such that,

$F_{tk}(1) = 5$, $F_{tk}(2) = 5$ and $F_{tk}(3) = 5$

$$I_{11} = \frac{1}{3}\left\{ \underset{T_{k1}}{\underset{\uparrow}{\frac{1}{5}}} + \underset{T_{k2}}{\underset{\uparrow}{\frac{1}{5}}} + \underset{T_{k3}}{\underset{\uparrow}{\frac{1}{5}}} \right\} \tag{2}$$

I_2 **Calculation:** The Token influence score for I_2 can be computed by using Eq. (3).

$$I_{2j} = \frac{|S_j|}{\sum_{t_k \in S_j} \frac{1}{F_{tk}(k)}} \tag{3}$$

For illustration, the token influence score for the 2nd token corresponding to 1st sentence can be modelled as per Eq. (4).

$$I_{21} = \frac{3}{5+5+5} = 3/15 \tag{4}$$

Each token has frequency $F_{tk}(k)=5$

I_3 **Calculation:** The Token influence score for I_3 can be computed by using Eq. (5).

$$I_{3j} = \frac{1}{|S_j|} \sum_{t_k \in S_j} \frac{F_{tk}(k)}{|T|} \tag{5}$$

Eq.(5) can be written as per Eq. (6).

$$I_{3j} = \frac{1}{|S_j||T|} \sum_{t_k \in S_j} F_{tk}(k) \tag{6}$$

Thus, for 3rd token corresponding to 1st sentence can be expressed as per Eq. (6). Here, three tokens appear for 5 times in S_1.

$$I_{31} = \frac{1}{3 \times 5}(5+5+5) = \frac{15}{15} = 1 \tag{7}$$

(d) The selected sentence S for t depends on the distribution of Vocabulary words β.

Then, the identified token (topic) based on the token influence score is denoted as D_{topic}, which is subjected to mapping text to contextualized embeddings based on the OBSum.

V. Text Mapping to Contextualized Embeddings, Sentence Clustering and Sentence Selection

A. *Text Mapping to Contextualized Embeddings : Architecture of OBSum based Approach*

The sentence S in the identified topic D_{topic} is mapped to an "n-dimensional vector of real numbers". The OBSum is designed by the summarizer to "map the sentences to contextualized embeddings". The TRANFORM was designed with the intention of pre-training the deep bidirectional representations from the unlabeled text by conditioning together on the right and left contexts. Typically, the TRANFORM framework encloses three major parts: Input layer, TRANFORM encoder and output layer. For illustration: if there are two sentences, 'we went to the river bank' and 'I need to go to the bank to make a deposit'. In both these sentence the token 'bank' is common, when are mapped under similar category they becomes meaningless, therefore the nature of the word 'bank' need to be identified to make it domain specific with higher precision. The TRANFORM considers both the right and left context before mapping the topic of a sentence under a category. To achieve this accurate mapping mechanism, the weight as well as biases of "pre-trained model" TRANFORM is updated with SAPSO. Thus, the architecture is denoted as OBSum and it is shown in Fig.2.

Input layer: The input sequence is constructed for the model by means of building "auxiliary sentence and the task" is turned into sentence-pair. An input sequence represents the sequence of texts or simpler texts in a token sequence, among which the first token is [CLS] and it contains the "special classification embedding", while the other is the special token [SEP] and it is utilized for separating segments or denoting the end of the sequence.

Adaptive TRANFORM encoder: It is a "multi-layer bidirectional transformer encoder" that is modelled on the basis of the original implementations. It encapsulates "12 layers (Transformer blocks) and 12 self-attention heads". The extracted $D_{topic} = \{I_1, I_2, I_3, T_1, T_2, ..T_N\}$ from multi-documents are the input given to the encoder.

Output layer: It is a simple "softmax classifier" and it is found above the Proposed TRANFORM encoder. The probability of the tokens in the sentence of D_{topic} is defined as per Eq. (8). In which, H is the final hidden state and G is the task-specific parameter matrix.

$$P(T|H) = Softmax(G.H) \tag{8}$$

The weight and bias factors being stable in the pre-trained model (TRANFORM); it is bit complex to process the natural languages of any data scale with higher accuracy. Such that, to make the topic categorization more precise, the bias (B) and the weight (W) in the TRANFORM model is fine-tuned with SAPSO.

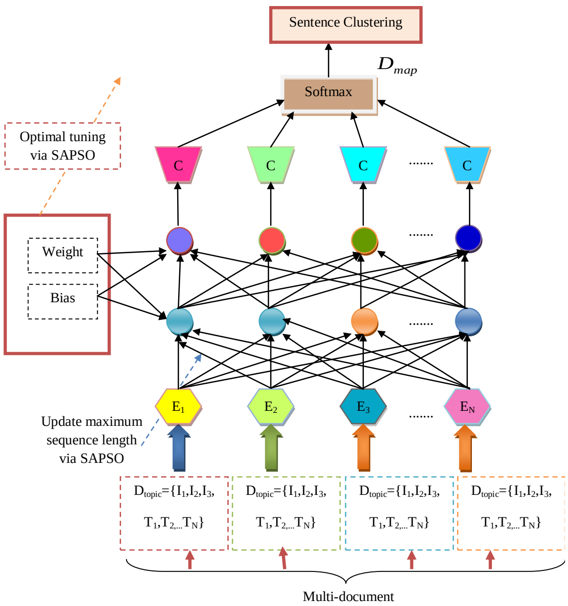


Fig. 2. Proposed OBSum Framework

B. Objective function and Solution Encoding

The objective function defined in this work is given in Eq. (9) and here Err is the "difference between the predicted and the actual outcomes". To achieve this objective Ob, the weight (W) and the bias (B) of the OBSum model is fine-tuned by the SAPSO. The input solution to the proposed algorithm is shown in Fig 3.

$$Ob = \min(Err) \tag{9}$$

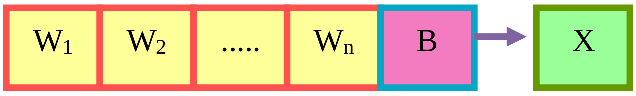


Fig. 3. Solution Encoding

C. SAPSO

PSO [31] is a swarm based optimization algorithm that is good in solving complex problems and finding the optimal solution on the basis of the behaviour of the flocking of the birds. On the other hand, the existing PSO suffers from lower convergence and hence there is a necessity of developing a new optimization algorithm. Therefore, a Self Adaptive PSO is introduced in this research work. The steps followed in the proposed SAPSO are described below:

Step 1 (Initialization): The population of search agents (Swarms) Pop is initialized along the $D-$dimensional space in a uniform manner. The maximal generation of the search agent is denoted as Max_{gen} and the current iteration is denoted as $iter$.

Step 2: Compute fitness for the search

Step 3: While $iter > Max_{gen}$, the each g particle move randomly with a "certain velocity" in the search space to find the global best position.

Step 4: Find the global best $P^t_{g(best)}$ and personal best P^t_{gd} for g particles in d^{th} dimension.

Step 5: The velocity and the position of the search agents are updated in traditional PSO by using two expressions. These were not found to be flexible for the randomly varying functions. Therefore, a new update expression is introduced in this research work based on three major conditions:

Condition 1: if $if\ r_1 < m_1$, then update the velocity V of search agents using Eq. (10) and Eq. (11)

$$V^{t+1}_{gd} = V^t_{gd} + c_1 r_1 \left(P^t_{gd} - X^t_{gd}\right) + c_2 r_2 \left(P^t_{g(best)} - X^t_{gd}\right) \tag{10}$$

$$X^{t+1}_{gd} = X^t_{gd} + V^{t+1}_{gd};\ g = 1,2,..,pop; d = 1,2,..,D \tag{11}$$

Where r_1 is a random number and m_1 is a mutation constant. In addition, X is the position of the search agent.

Condition 2: $if\ m_1 \le r_1 \le m_2$, then update the position of the search agent with the aid of the lion algorithms mutation mechanism. It is shown in Eq. (12), in which j genes. [32]

$$X^{new}(j) = X^{new}(j) + 0.3\mu(X_{\min}, X_{\max}).\xi \tag{12}$$

Condition 3: $if\ r_1 > m_2$, then compute mean μ of Minimum $X_{\min}$ and maximum $X_{\max}$ limits of every gene value as per Eq. (13)[32].

$$X^{new}(j) = \mu(X_{\min}, X_{\max}) \tag{13}$$

The pseudo code of the proposed SAPSO model is shown in Algorithm 1.

Algorithm 1: Pseudo code of SAPSO
Initialize Pop, Max_{gen}, $iter$
Compute fitness for the search
While $iter > Max_{gen}$
Find the global best $P^t_{g(best)}$ and personal best P^t_{gd} for g particles in d^{th} dimension.
if $if\ r_1 < m_1$
update the velocity V of search agents using Eq. (10)
else $if\ m_1 \le r_1 \le m_2$
update the position of the search agent using Eq.(12)
else $if\ r_1 > m_2$
update the position of the search agent using Eq.(13)
End if
End

D. Sentence clustering

The clustering step is utilized by the summarizer to group the similar sentences into groups as per the distance of their representations in the vector space. "An agglomerative hierarchical clustering procedure" is utilized in this research work to formulate clusters for the D_{map} sentences. The relatedness of sentences can be assessed by diverse measures within the clustering algorithm [1]. In a vector space, the most commonly utilized measures are the Euclidean distance, which considers the magnitude of the vector in different dimensions and cosine similarity that deals with the direction of the vector. The Euclidean distance ED and the Cosine similarity can be computed between two vectors $A = \{a_1, a_2,.., a_N\}, B = \{b_1, b_2,.., b_N\}$ as per Eq. (14) and Eq. (15), respectively.

$$ED(A,B) = \sqrt{\sum_{i=1}^{N} (a_i - b_i)^2} \tag{14}$$

$$CS(A,B) = \frac{A.B}{\|A\|.\|B\|} \tag{15}$$

E. Sentence Selection

The identified domain specific topics in a sentence are grouped together in this phase. The summarizer considers a wide range of important contents to generate the summary [1]. The summary is produced by means of selecting the "informative sentences" from all the clusters.The count of sentences to be selected is specified by the summarizer from each cluster as defined in Eq. (16)

$$N_i = N\frac{|C_i|}{|D_{map}|} \tag{16}$$

Where, the count of sentence that is extracted by the summarizer from clusters C_i is denoted as N_i and the quantity of m sentences selected for inclusion in the summary is denoted as N. In addition, the size of the cluster is denoted as $|C_i|$. The summarizer computes the informativeness score In_{score} to have a shared content between sentences in the cluster. The informativeness score is computed for the sentence S_i corresponding to cluster C_j by averaging the relatedness values between S_i and the whole sentence S_q in C_j. This is mathematically shown in Eq. (17), in which $relatedness(S_{i,}S_q)$ is the value in the form of cosine similarity (CS or Euclidean distance (ED) between the corresponding vectors, such that $i \neq q$.

$$In_{score} = \begin{cases} 1/|C_j|\sum_{q=1}^{|C_j|} relatedness(S_{i,}S_q) & if\ CS\ \ is\ used \\ \dfrac{1}{1/|C_j|\sum_{q=1}^{|C_j|} relatedness(S_{i,}S_q)} & if\ ED\, is\, used \end{cases} \tag{17}$$

The sentence clustering measure is equivalent to the measure utilized for relatedness value computation. On the basis of the informativeness score, the sentences are ranked by the summarizer within each of the clusters. If a sentence is said to achieve the highest informativeness scores, then it is said to be more reliable as it is said to share the most content within the same cluster. Finally, the sentences are put together by the summarizer and then it is sorted on the basis of their aappearance order in the original document. The generated summary is denoted as $D_{summary}$.

VI. Results and Discussion

A. Simulation procedure

The proposed text headline generation based on domain – specific topic modelling with TRANFORM+ SAPSO was implemented in PYTHON and the corresponding outcomes acquired are noted. The evaluation is done with 4 sets of domains: "Opinosis, BBC news, DUC 2002 datasets, BBC sports". The dataset description of BBC news, BBC sports, Opinosis and DUC 2002 datasets are tabulated in Table III, Table IV, Table V and Table VI, respectively. The proposed TRANFORM+SAPSO model is compared over the existing models like TRANFORM [1], NN [4], LDA [5], and LSTM-CNN [6] in terms of convergence and ROUGE-1 (R1) and ROUGE (R2) as well.

TABLE III. Dataset Description of BBC News

Downloaded from	topics	Count of text documents	Count of words unique
http://mlg.ucd.ie/datasets/bbc.html in the year 2014-2015	business, entertainment , politics, sport and tech	2225	9636

TABLE IV. Dataset Description of BBC Sports

Downloaded from	topics	Count of text documents	Count of words unique
https://www.bbc.com/sport in the year 2014-2015	athletics, cricket, football, rugby 737and tennis	737	4613

TABLE V. Dataset Description of Opinosis

Downloaded from	topics	'opinion seeking' queries constructed with	Count of queries compiled	Count of sentences / review document
Reviews collected from Tripadvisor, Amazon, Edmunds : https://www.mturk.com	hotels, cars and various product	2 humans in terms of entity name and a topic of interest.	51	100

TABLE VI. Dataset Description of DUC 2002

dataset	topics	Cluster	Doc.	Ref.	Limitation
DUC 2002 [33]	Biography, Culture, Business, Health, Politics, Law, Society, Natural Disaster, Science, Sports and International	59	567	116	100 words

B. Convergence Analysis

The objective of the current research work lies in maximizing the text headline generation accuracy. To prove that the objective being specified is reached, the convergence analysis is carried out. Fig. 4 shows the convergence analysis for Opinosis (in Fig.4 (a)), BBC English news (in Fig.4 (b)), DUC 2002 datasets (in Fig.4(c)), BBC sports (in Fig.4 (d)), respectively. This evaluation is done by varying the count of iterations. Here, X-axis denotes the count of iterations and Y-axis denotes the acquired cost function. The convergence for all 4 datasets (Opinosis, BBC English news, DUC 2002 datasets, BBC sports) is initially higher at the lower count of iterations and when the count of iterations is increased, there is a clear view that the convergence is lessened. For Opinosis, the convergence of the presented work at 100th iteration is 40% and 25% better than the existing models like TRANFORM and LSTM- CNN, respectively. Similar to this, the convergence of the presented and the traditional work lies between the ranges 3-4 for BBC English news headline generation in Fig. 4(d), but the presented work is converged to the minimal value that reaches the objectives defined in this work. Alike this, for BBC sports, the presented work has attained less convergence rate and hence achieves the objective of error minimization.

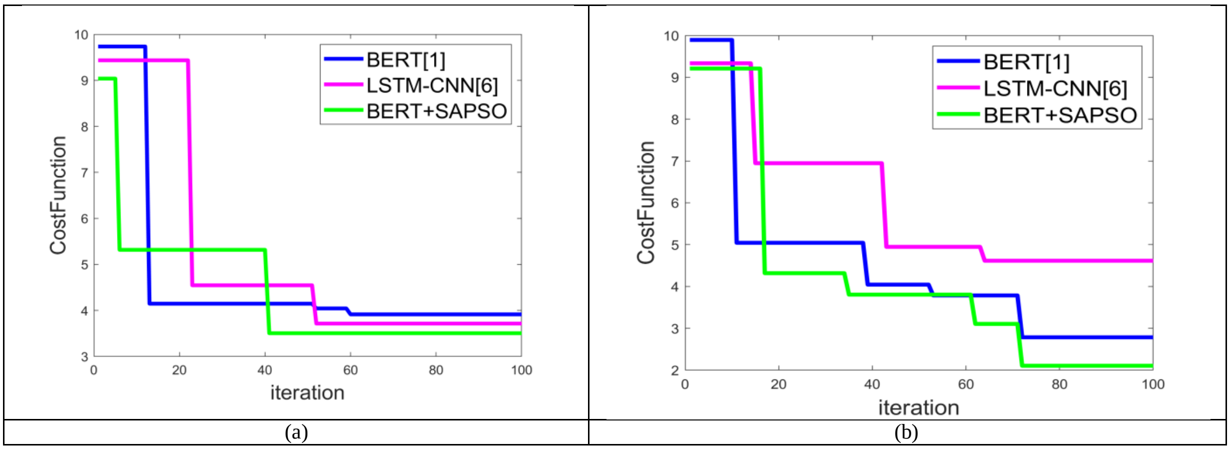


Fig. 4. Convergence analysis of the presented work and existing work for (a)RNN, (b)TRANSFORM

C. Analysis on ROUGE-1 and ROUGE-2

The presented text headline generation with domain – specific topic modelling with TRANFORM+ SAPSO is compared over the existing models in terms of ROUGE-1 (R1) and ROUGE-2 (R2), respectively. Here, the evaluation is done by varying the data rate. The resultant acquired for Opinosis is shown in Fig.5. Here, the R1 of the presented work at the data rate of 0.8 is 4.105, 2.7%, 0.68% and 1.255 better than the existing models like TRANFORM, NN, LDA and LSTM-CNN, respectively. In addition, in case of R2, the presented work is 6.6%, 3.3%, 2.4% and 2% better than the existing models like TRANFORM, NN, LDA and LSTM-CNN, respectively. For R2, at data rate 0.4, the presented work achieves the R2 value as 0.2754, while the R2 of the existing works are TRANFORM= 0.2709, NN= 0.2714, LDA= 0.2727 and LSTM-CNN= 0.2731.Thus, it is clear that the presented work has the highest R1 and R2 scores for every variation in the data rate.

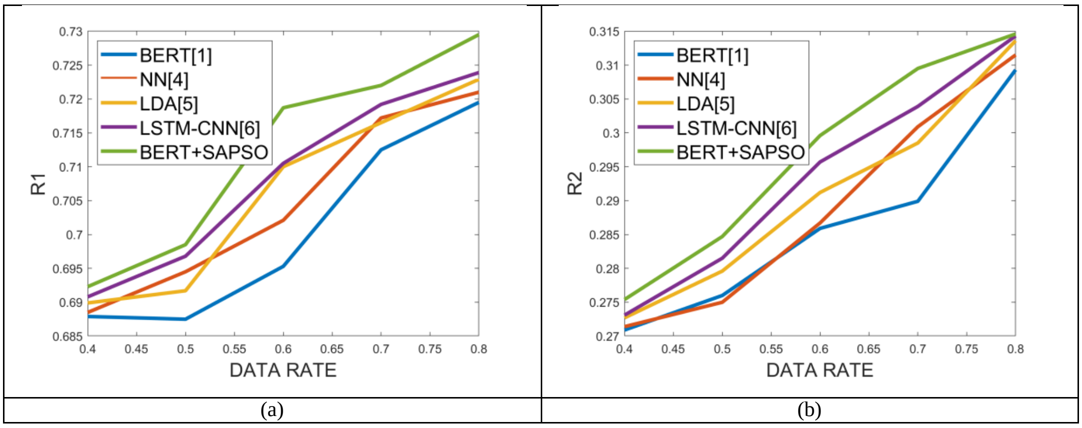
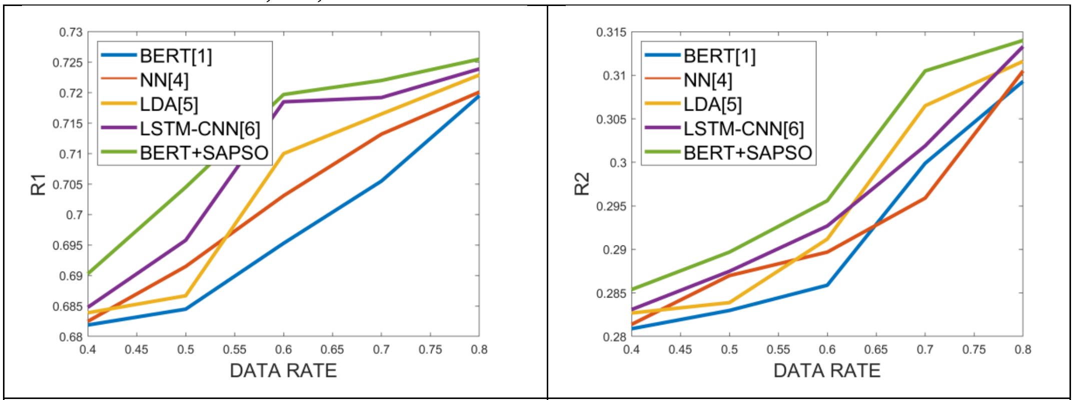


Fig. 5. Analysis on presented and existing works for varying data rate of Opinosis dataset showing the resultant of (a) R1 and (b) R2

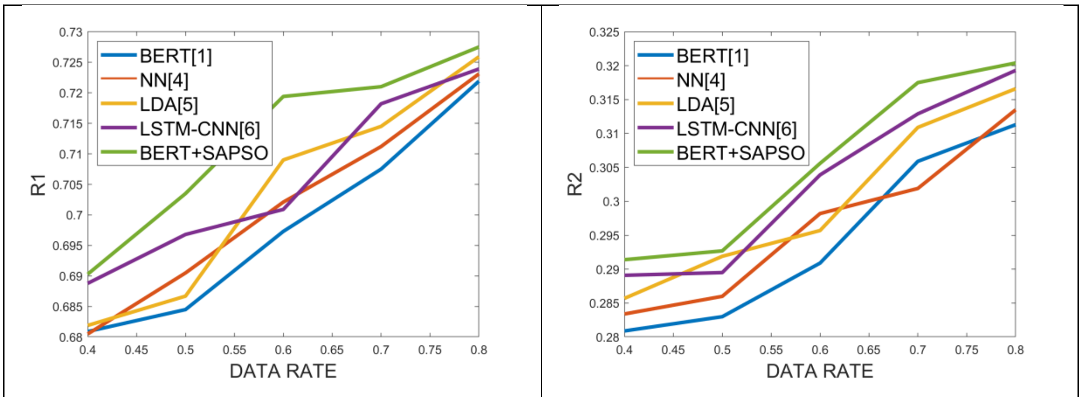
On the other hand, the R1 and R2 scores of BBC English news are shown graphically in Fig.6. On observing the graph, there is a clear view that the presented work has the highest R1 and R2 scores for every variation in the data rate. Initially, the R2 of BBC English news is lower and it tends to increase with the increase in the data arte. At the highest data rate of 0.8, the presented work shows the maximal R1 score as 0.725. Then, the R1 score of the presented work is 0.722 at data rate =0.7, while the R2 values of the TRANFORM, NN, LDA and LSTM-CNN are 0.7055, 0.7132, 0.7165 and 0.7192, respectively. In addition, the R2 of the presented work is also higher than all the existing works like TRANFORM, NN, LDA and LSTM-CNN, respectively for every variation in the data rate. Thus, as a whole it is clear that the presented works shows the highest R1 and R2 scores for every variation in the data rate.



(a) (b)

Fig. 6. Analysis on presented and existing works for varying data rate of BBC English news dataset showing the resultant of (a) R1 and (b) R2

Further, the R1 and R2 scores for DUC 2002 datasets is shown in Fig.7. The R1 achieves the highest value as 0.72 at the data rate of 0.6 and R2 reaches the maximal value as 0.295 at the data rate of 0.7. Apart from this, for every variation in data rate the presented work shows the highest R1 and R2 values.



(a) (b)

Fig. 7. Analysis on presented and existing works for varying data rate of DUC 2002 dataset showing the resultant of (a) R1 and (b) R2

In addition, the R1 and R2 scores corresponding to the BBC sports dataset is shown in Fig.8. Here, the R1 of the presented work at the data rate of 0.7 is 28.5%, 20%, 18.25% and 16.2% better than the existing models like TRANFORM, NN, LDA and LSTM-CNN, respectively. In addition, in case of R2, the presented work is 26.6%, 13.3%, 12.4% and 10% better than the existing models like TRANFORM, NN, LDA and LSTM-CNN, respectively. Thus, from the evaluation it is clear that the presented works exhibits highest R1 and R2 scores for every variation in the data rate.

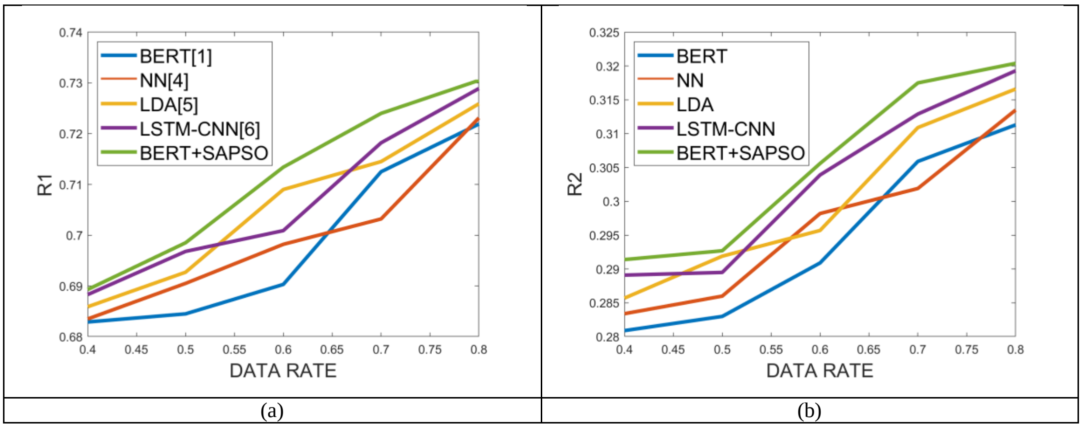


Fig. 8. Analysis on presented and existing works for varying data rate of BBC sports dataset showing the resultant of (a) R1 and (b) R2

VII. Conclusion

In this paper, a novel text headline generation approach with domain specific topic modeling is introduced by following five major stages: "Pre-processing, topic identification via tokenization, mapping text to contextualized embeddings, sentence clustering, and sentence selection". The collected raw DUC 2002 data were pre-processed and the domain –specific topic are identified from it using the token influence score based tokenization approach. The OBSum was designed in this research work to map the sentences to contextualized embeddings. In OBSum framework**,** the weight and biases of TRANFORM were fine-tuned, which an improved version of PSO. In addition, the maximal sequence length of TRANFORM. These generated contextualized embeddings were subjected to sentence clustering. At the end, the summary was generated by selecting informative sentences from all the clusters. The proposed text headline generation was compared over the existing models in terms of certain performance measures lie ROUGE 1, ROUGE 2 as well. The R1 of the presented work for DUC 2002 data, at the data rate of 0.7 is 28.5%, 20%, 18.25% and 16.2% better than the existing models like TRANFORM, and LSTM-RNN, respectively.

References

[1] Milad Moradi, Georg Dorffner, Matthias Samwald,"Deep contextualized embeddings for quantifying the informative content in biomedical text summarization",Computer Methods and Programs in Biomedicine, 2019

[2] Jinming Zhao, Ming Liu,Longxiang Gao,Yuan Jin,Lan Du and He Zhao,"SummPip: Unsupervised Multi-Document Summarization with Sentence Graph Compression",SIGIR,2020

[3] Akanksha Joshi, E. Fidalgo,E. Alegre,Laura Fernández-Robles,"SummCoder: An unsupervised framework for extractive text summarization based on deep auto-encoders",Expert Systems With Applications,vol.129,pp.200-215,2019

[4] Deepa Anand and Rupali Wagh,"Effective deep learning approaches for summarization of legal texts",Journal of King Saud University – Computer and Information Sciences,2019

[5] Qasem A. Al-Radaideh and Dareen Q.Bataineh, " A Hybrid approach for Arabic text summarization Using Domain Knowledge and Genetic algorithms", Cognitive Computation, March, 2018

[6] Shengli Song,Haitao Huang & Tongxiao Ruan ,"Abstractive text summarization using LSTM-CNN based deep learning",Multimedia Tools and Applications , vol.78,pp.857-875,2019

[7] Nabil Alami, Noureddine En-nahnahi, Said Alaoui Ouatik & Mohammed Meknassi ,"Using Unsupervised Deep Learning for Automatic Summarization of Arabic Documents",Arabian Journal for Science and Engineering , vol.43, pp.7803-7815,2018

[8] Minakshi Tomer & Manoj Kumar,"Improving Text Summarization using Ensembled Approach based on Fuzzy with LSTM",Arabian Journal for Science and Engineering,2020

[9] Zhenrong Deng, Fuxin Ma, Rushi Lan, Wenming Huang, Xiaonan Luo,"A Two-stage Chinese text summarization algorithm using keyword information and adversarial learning",Neurocomputing, in communication, 2020

[10] Asad Abdi, Siti Mariyam Shamsuddin, Shafaatunnur Hasan, Jalil Piran,"Machine learning-based multi-documents sentiment-oriented summarization using linguistic treatment",Expert Systems with Applications,2018

[11] Nabil Alami, Mohammed Meknassi, Noureddine En-nahnahi,"Enhancing unsupervised neural networks based text summarization with word embedding and ensemble learning",Expert Systems with Application,2019

[12] Arturo Curiel, Claudio Gutiérrez-Soto, José-Rafael Rojano-Cáceres,"An online multi-source summarization algorithm for text readability in topic-based search",Computer Speech & Language, in communication,2020

[13] X. Lin, M. Liu and J. Zhang, "A Top-Down Binary Hierarchical Topic Model for Biomedical Literature," in IEEE Access, vol. 8, pp. 59870-59882, 2020, doi: 10.1109/ACCESS.2020.2983265.

[14] Rupal Bhargava, Yashvardhan Sharma,"Deep Extractive Text Summarization",Procedia Computer Science,2020

[15] Shengluan Hou, Ruqian Lu,"Knowledge-guided unsupervised rhetorical parsing for text summarization",Information Systems,2020

[16] Rupal Bhargava, Gargi Sharma, Yashvardhan Sharma,"Deep Text Summarization using Generative Adversarial Networks in Indian Languages",Procedia Computer Science,2020

[17] Amy J. C. Trappey, Charles V. Trappey, Jheng-Long Wu, Jack W. C. Wang,"Intelligent compilation of patent summaries using machine learning and natural language processing techniques",Advanced Engineering Informatics,2020

[18] Begum Mutlu, Ebru A. Sezer, M. Ali Akcayol,"Candidate sentence selection for extractive text summarization",Information Processing & Management,2020

[19] Wafaa S. El-Kassas, Cherif R. Salama, Ahmed A. Rafea, Hoda K. Mohamed,"Automatic text summarization: A comprehensive survey",Expert Systems with Applications,2021

[20] Zeyu Liang, Junping Du, Chaoyang Li,"Abstractive social media text summarization using selective reinforced Seq2Seq attention model",Neurocomputing,2020

[21] Farooq Zaman, Matthew Shardlow, Saeed-Ul Hassan, Naif Radi Aljohani, Raheel Nawaz ,"HTSS: A novel hybrid text summarisation and simplification architecture",Information Processing & Management,2020

[22] Adhika Pramita Widyassari, Supriadi Rustad, Guruh Fajar Shidik, Edi Noersasongko, De Rosal Ignatius Moses Setiadi,"Review of automatic text summarization techniques & methods",Journal of King Saud University - Computer and Information Sciences,2020

[23] Min Yang, Xintong Wang, Yao Lu, Jianming Lv, Chengming Li,"Plausibility-promoting generative adversarial network for abstractive text summarization with multi-task constraint",Information Sciences,2020

[24] Jiyuan Zheng, Zhou Zhao, Zehan Song, Min Yang, Xiaohui Yan,"Abstractive meeting summarization by hierarchical adaptive segmental network learning with multiple revising steps",Neurocomputing,2020

[25] Duy Duc An Bui, Guilherme Del Fiol, John F. Hurdle, Siddhartha Jonnalagadda,"Extractive text summarization system to aid data extraction from full text in systematic review development",Journal of Biomedical Informatics,2016.

[26] M. Yang, C. Li, Y. Shen, Q. Wu, Z. Zhao and X. Chen, "Hierarchical Human-Like Deep Neural Networks for Abstractive Text Summarization," IEEE Transactions on Neural Networks and Learning Systems,2020.

[27] Rajakumar B R, “Optimization using lion algorithm: a biological inspiration from lion’s social behavior”, Evolutionary Intelligence, Special Issue on Nature inspired algorithms for high performance computing in computer vision, Vol. 11, No. 1-2, pages 31-52, 2018, DOI: https://doi.org/10.1007/s12065-018-0168-y.

[28] B. R. Rajakumar, "Impact of Static and Adaptive Mutation Techniques on Genetic Algorithm", International Journal of Hybrid Intelligent Systems, Vol. 10, No. 1, pages: 11-22, 2013, DOI: 10.3233/HIS-120161

[29] B. R. Rajakumar, "Static and Adaptive Mutation Techniques for Genetic algorithm: A Systematic Comparative Analysis", International Journal of Computational Science and Engineering, Vol. 8, No. 2, pages: 180-193, 2013, DOI: 10.1504/IJCSE.2013.053087

[30] B. R. Rajakumar, “Lion algorithm for standard and large scale bilinear system identification: A global optimization based on Lion's social behavior”, 2014 IEEE Congress on Evolutionary Computation, Beijing, China, July 2014, pages: 2116-2123, DOI: 10.1109/CEC.2014.6900561

[31] M.R.Tanweer, S.Suresh, and N.Sundararajan, " Self regulating particle swarm optimization algorithm", Information Sciences, vol. 294, pp. 182-202, 2015.

[32] B. R. Rajakumar and Aloysius George, "A New Adaptive Mutation Technique for Genetic Algorithm", In proceedings of IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), pages: 1-7, December 18-20, Coimbatore, India, 2012, DOI: 10.1109/ICCIC.2012.6510293

[33] Ziqiang Cao, Wenjie Li,Sujian Li and Furu Wei,"Improving Multi-Document Summarization via Text Classification",Association for the Advancement of Artificial Intelligence (www.aaai.org), Nov 2016