Asian Journal of Applied Sciences (ISSN: 2321 – 0893)
Volume 04 – Issue 06, December 2016

Text Summary using Modified Particle Swarm Optimization Algorithm

V. S. Raj Kumar[1], R. Danu[2], S. Shanmugapriya[3,*], R. Vinod[4]

[1]Assistant Professor, Department of Computer Science and Engineering
Vel Tech High Tech Dr.Rangarajan and Dr.Sakunthala Engineering College, Avadi, Chennai-62, Tamilnadu, India.

[2]Assistant Professor, Department of Computer Science and Engineering
Vel Tech High Tech Dr.Rangarajan and Dr.Sakunthala Engineering College, Avadi, Chennai-62, Tamilnadu, India.

[3]Assistant Professor, Department of Computer Science and Engineering
Vel Tech High Tech Dr.Rangarajan and Dr.Sakunthala Engineering College, Avadi, Chennai-62, Tamilnadu, India.

[4]Assistant Professor, Department of Computer Science and Engineering
Vel Tech High Tech Dr.Rangarajan and Dr.Sakunthala Engineering College, Avadi, Chennai-62, Tamilnadu, India.

[*]*Corresponding author's email: spriyavasan [AT] yahoo.co.in*

ABSTRACT— ***Text summarization is the process of provides meaningful and short contents of the documents in the automated manner by using which concept of entire documents can be found. In the existing work, an approach called as unsupervised generic summary creation is introduced to summarize the single document and as well as multiple documents in the generic manner. However, the genetic algorithm will consume more computational time for generating the summaries in case of presence of multiple documents with more sentences. This problem is resolved in the proposed approach by introducing the modified particle swarm optimization where global best would be updated based on weighted mean approach. This proposed approach provides an efficient and flexible creation of summaries with reduced computation time. The experimental tests conducted were proves that the proposed approach provides better result than the existing approach in terms of reduced computational time.***

Keywords— Differential Evolution Algorithm, Modified Particle Swarm Optimization Algorithm

1. INTRODUCTION

Automatic Text summarization is the process of generating a concise version of a text that contains the important information which is presented in the original documents. Text summarization can be divided into two types namely, single-document and a multi-document summarization. A Single-document summarization is the process of condensing a single document into a shorter version. A multi-document summarization is the process of condensing a set of documents into a concise form.

In multi-document summarization, simply selecting the sentences with high relevance score may lead to the redundancy problem. By focusing on both the relevance scores of sentences and the sentence to sentence relation, the redundancy problem can be solved. In this paper, content coverage and redundancy are simultaneously considered in the modified particle swarm optimization method.

2. LITERATURE SURVEY

Rasim M. Alguliyev et al. [1] proposed an optimization-based unsupervised approach to automatic document summarization to improve the performance of text summarization process. Maolong Xi et al. [2] proposed an improved quantum-behaved particle swarm optimization algorithm with weighted mean best position. S.A.Babar et al. [3] focuses on improving the performance of text summarization by using the Fuzzy logic Extraction approach to extract the sentences based on high relevance score and also based on the level of importance of the sentences.

You Ouyang et al. [4] focus on applying regression models to query-focused multi-document summarization by using SVR (Support Vector Regression). J. Tang et al. [5] proposed a multi-topic based query-oriented summarization to solve the multi-topic query. Rasim Alguliev et al. [6] suggested an evolutionary algorithm for creating the summary by extracting and clustering the sentences from the original document.

S.Prabha et al. [7] proposed a Context-Based Similarity Analysis for Document Summarization. The lexical association between terms is used to provide a context sensitive weight to the document terms. Sunita Sarkar et al. [8]

presented a comparative analysis to prove that hybrid PSO with K-means performs better than K-means and PSO algorithms. Chen Li et al. [9] focuses on the problem of using sentence compression techniques to improve multi-document summarization.

3. SYSTEM ARCHITECTURE

After preprocessing the source document, the similarity between the sentences is calculated by using semantic similarity measure. And then fitness value of maximum relevance and minimum redundancy is evaluated for the sentences. The summary is generated by using the differential evaluation algorithm and also by using modified PSO algorithm. In performance evaluation, the comparison between the results of both algorithms is done. Figure 1 depicts the architecture diagram of the proposed system.

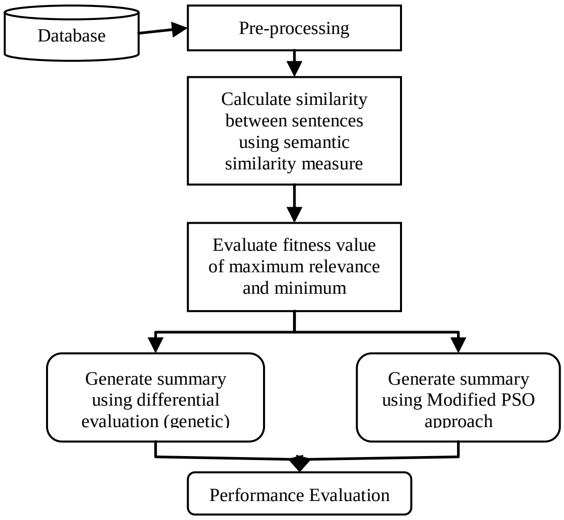

Figure 1: Architecture Diagram

4. METHODOLOGY

4.1 Preprocessing

In preprocessing, the documents are decomposed into a set of sentences and each sentence has a collection of terms. The stop words and special characters in each sentence are removed from the sentence collection.

4.2 Similarity Measure Between Sentences

Based on the ordering of sentences, similarity measure is divided into two types. Symmetric and asymmetric measures are the two types of similarity measures. In this module, the combination of both types of similarity measures is used to measure the similarity between the sentences.

4.3 Calculating Fitness Of Sentences

Each sentence s_i is associated with a positive weight of i, which is defined as following

$$\alpha_i = \frac{\mathrm{sml}(S_i, O)}{\sum_{j=1}^{n} \mathrm{sml}\,(S_j, O)}, \quad i = 1, \ldots, n$$

where O defines the center of the collection S.

4.4 Using Differential Evolution Algorithm for Summarization

Differential Evolution (DE) algorithm is established on the framework of genetic algorithms. DE algorithm has three operators namely crossover, mutation and selection which are similar to the genetic algorithms.

The proposed algorithm is

1. Creating an initial population.
2. Convert the real-valued chromosome into a binary space.
3. Calculating the fitness value of each chromosome in the population.

4. Selecting of best chromosome.
5. Generating an offspring.
6. For the component of mutant vector, check the boundary constraints.
7. Then convert the real-valued chromosome into the binary space and decode it to the problem variables.

4.5 Using Modified Particle Swarm Optimization (Pso) Algorithm For Summarization

Input: documents

Output: summarized document

1. Initialize N number of particles with set of tasks and allocate the resources randomly, a position of particle is denoted by x_i and velocity is denoted as V_i.
2. Particle position is initialized as x_i randomly in the search space.
3. Set the particle's best known position to its initial position

   $pbest \leftarrow x$
4. Initialize each particle's velocity v to random values.
5. Until a termination criterion is met, repeat the following:
6. For every particle i=1, 2… N
7. Compute the fitness value.
8. Pick a random vector $a \sim U(-d, d)$
9. Add this to the current solution x, to create the new potential solution y.
10. If $(f(y) < f(pbest)$ then update the particle's best known position.
11. Otherwise decrease the search range by multiplication factor q.
12. Compute the weighted mean value for gbest,

$$\text{Wgbest (t)} = (M_1(t), M_2(t), \ldots M_3(t)) = \left(\frac{1}{M}\sum_{j=1}^{M} b_{i,1}\, G_{j,1}^{t}, \frac{1}{M}\sum_{i=1}^{M} b_{i,2} G_{j,2}^{t}, \ldots\ldots\ldots \frac{1}{M}\sum_{i=1}^{M} b_{i,n}\, G_{j,n}^{t}\right)$$

4.6 Performance Evaluation

In this module performance evaluation were done to compare the effectiveness of the proposed methodology by comparing it with the existing approach in terms multiple performance measures. The performance measures are considered for the proving the improvement of the proposed work is the accuracy and the computational time.

5. RESULTS AND DISCUSSION

5.1 Loading Dataset

The dataset consists of collection of text documents. The 20_newsgroup which contains a collection of text documents is taken has input document. The size of the text documents various with each other in the given dataset.

5.2 Preprocessing

In this stage, the given document is preprocessed. The document is splitted into sentences based on the (.) separator. And then the special characters and stop words are removed from the sentences.

5.3 Similarity Calculation Result

The Figure 2 depicts the matrix format of similarity calculation result of each word with other words in the document.

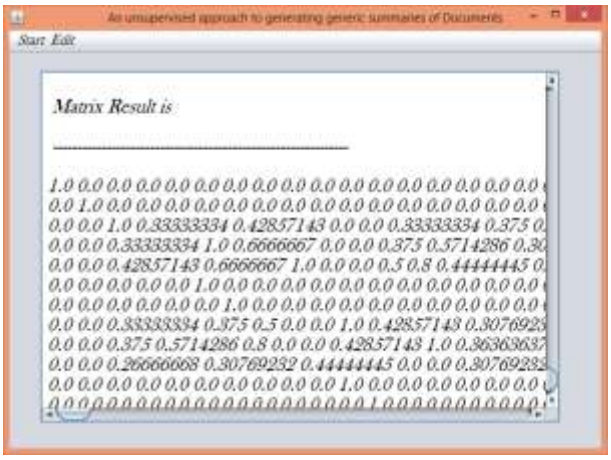

Figure 2: Similarity Calculation Result

5.4 Fitness Calculation

The fitness for each sentences and words are calculated. The Figure 3 depicts the fitness value for each sentence in a document.

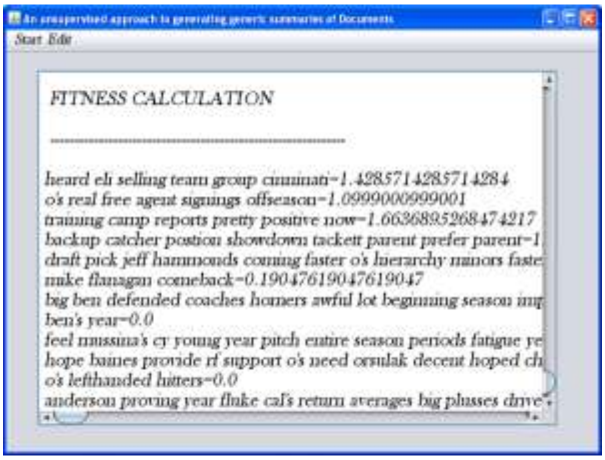

Figure 3: Fitness Calculation

5.5 Summarization Using Differential Evolution Algorithm

The Figure 4 depicts the fitness value of each distinct word present in the document.

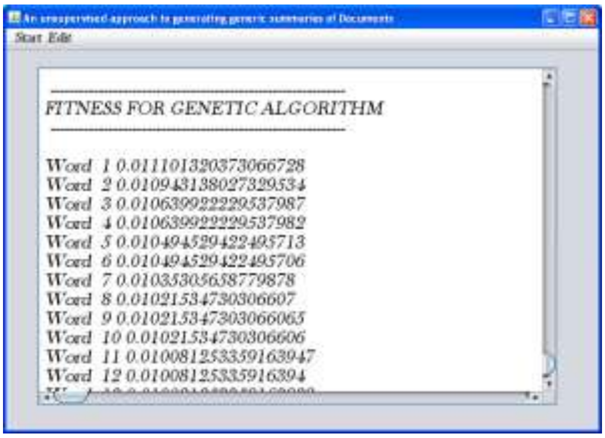

Figure 4: Fitness for Genetic Algorithm

Based on the fitness of each words, the words which contains the fitness value higher than the threshold is considered has an most important word that is related to the document.

5.6 Summarization Using Modifed Pso Algorithm

The text file containing matrix value is converted into .arff file format and then the file is taken as input. The number of folds value denotes the maximum folds to be generated in the given input data. Score per fold is generated.

While the classification type is set to MPSO, then the average score of each folds score is calculated. Based on the average score, the fold containing the score which is closer to the average score is selected has a best fold.

5.7 Performance Evaluation

Figure 5 given below contains the comparison chart to that the proposed MPSO algorithm has a high accuracy then the genetic and PSO algorithm.

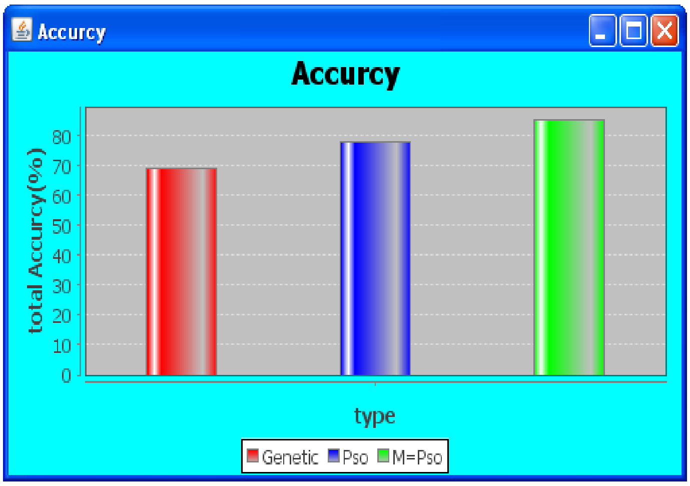

Figure 5: Performance Evaluation

6. CONCLUSION

In this work, the summarized output is generated by adapting the evolutionary algorithms which can select most optimal set of sentences that represents the meaning of the whole documents. This method also used to reduce the computational time in the case of multiple documents with more sentences. The experimental tests conducted were proves that the proposed approach leads to the better result in terms of improved summarized content.

7. REFERENCES

[1] Rasim M. Alguliyev, Ramiz M. Aliguliyev, Nijat R. Isazade (2015), "An unsupervised approach to generating generic summaries of documents", Applied Soft Computing 34236–250.

[2] Maolong Xi, Jun Sun, WenboXu (2008), "An improved quantum-behaved particle swarm optimization algorithm with weighted mean best position", Applied Mathematics and Computation 205, 751–759.

[3] S.A.Babar, Pallavi D.Patil (2014), "Improving Performance of Text Summarization", International Conference on Information and Communication Technologies (ICICT).

[4] Y. Ouyang, W. Li, S. Li, Q. Lu (2011), "Applying regression models to query-focused multi-document summarization", Information Processing and Management 47 (2) 227–237.

[5] J. Tang, L. Yao, D. Chen (2009), "Multi-topic based query-oriented summarization", in: Proceedings of the 9th SIAM International Conference on Data Mining, Nevada, USA, pp. 1148–1159.

[6] Rasim Alguliev, Ramiz Aliguliyev, "Evolutionary algorithm for extractive text summarization", Intelligent Information Management, 2009, 1, 128-138.

[7] S.Prabha, Dr.K.Duraiswamy, B.Priyanga, "Context-Based Similarity Analysis for Document Summarization", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 3, Issue 4, April 2014.

[8] Sunita Sarkar , Arindam Roy and B. S. Purkayastha, "A Comparative Analysis of Particle Swarm Optimization and K-means Algorithm For Text Clustering Using Nepali WordNet", International Journal on Natural Language Computing (IJNLC), Vol. 3, No.3, June 2014.

[9] Chen Li, Yang Liu, Fei Liu, Lin Zhao, Fuliang Weng, "Improving Multi-documents Summarization by Sentence Compression based on Expanded Constituent Parse Trees" , Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 691–701, October 25-29, 2014.