

Fuzzy Swarm Based Text Summarization

¹Mohammed Salem Binwahlan, ¹Naomie Salim and ²Ladda Suanmali
¹Faculty of Computer Science and Information Systems,
University Technology Malaysia, 81310, Skudai, Johor, Malaysia
²Faculty of Science and Technology, Suan Dusit Rajabhat University,
Dusit, Bangkok, 10300, Thailand

Abstract: Problem statement: The aim of automatic text summarization systems is to select the most relevant information from an abundance of text sources. A daily rapid growth of data on the internet makes the achieve events of such aim a big challenge. **Approach:** In this study, we incorporated fuzzy logic with swarm intelligence; so that risks, uncertainty, ambiguity and imprecise values of choosing the features weights (scores) could be flexibly tolerated. The weights obtained from the swarm experiment were used to adjust the text features scores and then the features scores were used as inputs for the fuzzy inference system to produce the final sentence score. The sentences were ranked in descending order based on their scores and then the top n sentences were selected as final summary. **Results:** The experiments showed that the incorporation of fuzzy logic with swarm intelligence could play an important role in the selection process of the most important sentences to be included in the final summary. Also the results showed that the proposed method got a good performance outperforming the swarm model and the benchmark methods. **Conclusion:** Incorporating more than one technique for dealing with the sentence scoring proved to be an effective mechanism. The PSO was employed for producing the text features weights. The purpose of this process was to emphasize on dealing with the text features fairly based on their importance and to differentiate between more and less important features. The fuzzy inference system was employed to determine the final sentence score, on which the decision was made to include the sentence in the summary or not.

Key words: Fuzzy logic, membership function, particle swarm optimization, summarization, text feature

INTRODUCTION

The aim of automatic text summarization systems is to select the most relevant information from an abundance of text sources. A daily rapid growth of data on the internet makes the achieving of such aim as a big challenge. To overcome that challenge, much work is still required and may be it is beyond of only one technique. Therefore in this study, we investigate the incorporation of fuzzy logic with swarm intelligence. In automatic text summarization where the sentence score is based on the weights of the features, choosing those weights can be imprecise and uncertain, by the incorporation of fuzzy logic with swarm intelligence, so that risks, uncertainty, ambiguity and imprecise values can be flexibility tolerated.

Automatic text summarization researchers since Luhn's research^[1], they are trying to solve or at least relieve the challenge by proposing techniques for

generating summaries. The summaries serve as quick guide to the interesting information, providing a short form for each document in the document set; reading summary makes decision about reading the whole document or not and it also serves as time saver.

The machine learning approaches^[2-8] have proven their ability in improving the summarization performance. In our previous study^[9], we used Particle Swarm Optimization (PSO)^[10] as machine learning for features selection problem in order to study the feature structure effect on the feature selection, one of the results obtained from that study is the learned features weights. In this research, we will try to apply the features weights produced by PSO in a proposed method for automatic text summarization problem. Based on our literature survey for applying fuzzy swarm for text summarization problem, we found that fuzzy swarm has not been introduced for such problem. PSO was successfully applied in some related problems

Corresponding Author: Mohammed Salem Binwahlan, Faculty of Computer Science and Information Systems, University Technology Malaysia, 81310, Skudai, Johor, Malaysia

like text classification and data clustering and could achieve high performance. Ziegler and Skubacz^[11] proposed a method for extracting the text content from news web pages, where the web pages are in HTML format, particle swarm optimization was used to learn features thresholds from training set, the features having score greater or equal PSO-determined threshold are counted for their text to be classified as signal text and the features having score less than such threshold are not counted to their text and due to the later, the text may be classified as noise. The signal text is extracted as text content. Merwe and Engelbrecht^[12] introduced PSO based clustering algorithm, where each particle in PSO represents a different group of centroids. Also in the same study, a hybrid PSO and K-means as clustering algorithm was investigated, where the K-means is run first and its results (clusters centroids) are used for initialing one particle in PSO while the remaining particles are initialized randomly. Cui *et al.*^[13] have applied PSO for documents clustering. The particle components are same as they were used in^[12], where each particle consists of a number of centroids. The researchers have taken into account execution time of PSO compared with K-means which has less execution time in the dealing with large data set, so they introduced a hybrid of PSO and K-means, where the role of PSO is to determine the initial centroids for K-means algorithms. Wang *et al.*^[14] presented PSO for web document classification, the highest weight terms were used as feature for the classification. The PSO was adopted for the classification problem, where each particle represents those highest weight terms. F-mean measure for precision and recall was used as fitness function.

Applying the fuzzy logic for text summarization still needs more investigation; a few studies were done in this direction, here we present some works which used fuzzy IF-THEN rules for scoring the sentences, Kiani-B and Akbarzadeh-T^[15] presented text summarization system in which the features are used as input for the fuzzy system, based on the fuzzy rules each sentence receives score in the range between zero and one, the fuzzy rules were optimized using hybrid GA and GP. Kyoomarsi *et al.*^[16] proposed fuzzy logic based text summarization, following Kiani-B and Akbarzadeh-T's way^[15], the difference between these two studies is in the later, the fuzzy rules were not optimized.

The good performance of PSO and fuzzy logic in above studies promises that their integration in single method can do well for automatic text summarization as well.

MATERIALS AND METHODS

Particle Swarm Optimization (PSO): Particle swarm optimization^[10] originally relates to artificial life (A-life) in general and specifically it connects with bird flocking and fish schooling^[17]. The Intelligence in PSO as any other swarm technique is a collective intelligence resulting in the collective behaviors of (unsophisticated) individuals interacting locally and with their environment causing coherent functional global patterns to emerge. Particle Swarm Optimization (PSO) which is inspired by the social behavior of bird flocking or fish schooling and Ant Colony Optimization (ACO) which is inspired by behavior of ants are the primary computational parts of swarm intelligence.

Particle swarm optimization was introduced by^[10] as a stochastic, population-based evolutionary algorithm for problem solving. The key idea of PSO method is to simulate the shared behavior happening among the birds flocks or fish school.

PSO depends on its methodology, a population of individuals to discover favorable regions of the search space. Every member in the population is called particle and the group of all particles is called a swarm. Each particle flies in the search space with a velocity that is dynamically adjusted according to its own flying experience and its companions' flying experience and retains the best position it ever encountered in memory. The best position ever encountered by all particles of the swarm is also announced to all particles. The study of PSO starts by initially randomizing a group of solutions (particles), the swarm updates its best value every cycle based on (1) and (2) and then after several iterations finds the optimized solution:

$$V_{id}(t+1) \leftarrow w * V_{id}(t) + c_1 r_1 (p_{id}(t) - x_{id}(t)) + c_2 r_2 (p_{gd}(t) - x_{id}(t)) \quad (1)$$

Where:

- $V_{id}(t)$ = The velocity of the particle i in the time point t in the search space along the dimension d
- $P_{id}(t)$ = The best position in which the particle previously got high fitness value, it is called pbest
- $x_{id}(t)$ = The current position of the particle i in the search space
- r_1 and r_2 = Random generated numbers in the range [0,1]
- $p_{gd}(t)$ = The overall best position in which a particle got best fitness value, it is called the gbest
- c_1 and c_2 = Acceleration parameters
- W = Inertia weight, its value is decreased linearly over the time from 0.9-0.4^[18].

$$x_{id}(t+1) \leftarrow x_{id}(t) + V_{id}(t+1) \quad (2)$$

Where:

$X_{id}(t+1)$ = The new position which the particle must move to
 $x_{id}(t)$ = The current position of the particle
 $V_{id}(t+1)$ = The new velocity of the particle resulting in the calculation in (1) which mainly determines the new position of the particle

The velocity of the particle must be in the range $[V_{max}, V_{min}]$.

There are two types of PSO: Continuous particle swarm optimization which is to optimize continuous nonlinear problems^[10] and binary particle swarm optimization^[19] which is extension of continuous PSO, in which the particle position is represented as bit string rather than real numbers; the update of the position in continuous PSO is done directly by adding the velocity to the previous position but in binary PSO, the velocity is used only in the sigmoid function as in (3) to calculate the probability of the bit value to be changed to 1 or 0, where the value retrieved from the sigmoid function is compared with random generated value in the range between zero and one:

$$x_{ij}(t+1) = \begin{cases} 0 & \text{if } p_{ij}(t) \geq \frac{1}{1 + \exp(-v_{ij}(t))} \\ 1 & \text{otherwise} \end{cases} \quad (3)$$

Text Features: The features used in this study are five^[9]:

- **Sentence Centrality:** The sentence centrality consists of three features: The similarity, shared friends and shared n-grams between the sentence in hand and all other document sentences, normalized by n-1, n is the number of sentences in the document
- **Title feature:** This feature is formed as average of two features which are title-help sentence (THS): The sentence containing n-gram terms of title and title-help sentence relevance sentence (THSRS): The sentence containing n-gram terms of any title-help sentence
- **Word sentence score (WSS):** It is calculated as the following:

$$WSS(S_i) = 0.1 + \frac{\sum_{t_j \in S_i} W_{ij}}{HTFS} \quad (4)$$

|no. of sentences containing $t_j \geq \frac{1}{2}LS$

Where:

0.1 = Minimum score the sentence gets in case it's terms are not important

W_{ij} = As in (5) is the term weight (TF-ISF) of the term t_{ij} in the sentence s_i

LS = Summary length and HTFS is highest term weights (TF-ISF) summation of a sentence in the document:

$$W_{ij} = tf_{ij} \times isf = tf(t_{ij}, s_i) \left[1 - \frac{\log(sf(t_{ij}) + 1)}{\log(n + 1)} \right] \quad (5)$$

- **Key word feature:** The top 10 words whose high TF-ISF score are chosen as key words
- **The similarity to first sentence:** This feature is to score the sentence based on its similarity to the first sentence in the document, where in news article, the first sentence in the article is very important sentence

Fuzzy logic: The term "fuzzy logic" resulted in the development of the theory of fuzzy sets by Zadeh^[20]. Due to the limitation of classic logic is that deals only with two values, true or false created the need for extending it to be able to handle the partial truth (neither completely true nor completely false). The fuzzy logic is extension of the classical logic in form of generalization of the classical logic inference rules (like modus ponens, modus tollens and hypothetical syllogism) which has ability to deal with approximate reasoning^[21]. The fuzzy set is an elaboration for the traditional set "crisp set" in which each member has a degree of membership to that set determined by membership function. The membership function is a function assigns membership degree to each member in the target set, the range of membership degree between zero and one. The computer can translate linguistic statement into actions based on a set of such IF-THEN rules of the fuzzy logic. The fuzzy IF-THEN rules are normally created as the form "if A then B" in which the condition is connected with actions, where A and B are fuzzy sets. The fuzzy logic has advantage in terms of simplicity of development and modification because the rules are well understandable and easy to modify, add new rules or remove existing rules.

The typical fuzzy inference system consists of the following stages:

Fuzzification: Fuzzification is a kind of uncertainty that requires fuzzy sets^[21], in which the input values are translated into grades of membership in the range between zero and one for linguistic terms of fuzzy sets using a membership function which is used to assign a grade to each linguistic term.

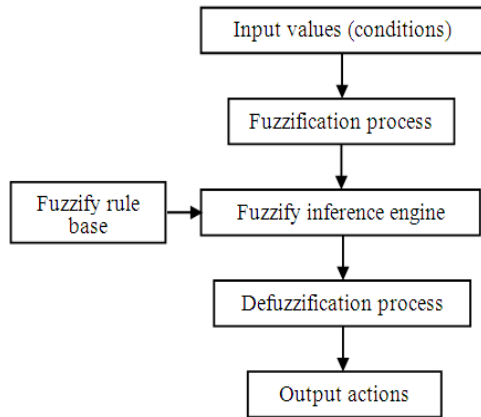


Fig. 1: Typical fuzzy system

Inference: The inference is the core part of a fuzzy system, which merges the facts obtained from the fuzzification part with a series of production rules to perform the fuzzy reasoning process^[21].

The most important fuzzy inference methods are Mamdani’s fuzzy inference method^[22] and Sugeno or Takagi-Sugeno-Kang method^[23]. The rule consequent in Mamdani fuzzy systems is represented using fuzzy sets, while in Takagi-Sugeno-Kang fuzzy systems; it is form as linear functions of input variables. Typical fuzzy system is shown in Fig. 1.

Defuzzification: The goal of defuzzification is to convert the fuzzy results of the inference into a crisp output.

The swarm based summarization: The swarm model is defined as combination of text features scores as in (6), where those features scores are adjusted using the weights resulting in the training of the Particle Swarm Optimization (PSO).

Therefore the first part in this model is for training PSO, 100 documents were selected from Document Understanding Conference (DUC)^[24] data collection, DUC 2002 and used as training and testing data. Figure 2-4 show the swarm model. The second part in this model is for testing the proposed model:

$$\text{Score}(s) = \sum_{i=1}^5 w_i \times \text{score_}f_i(s) \quad (6)$$

Where:

- Scor (s) = The score of the sentence s
- w_i = The weighted of the feature i produced by PSO, i = 1-5
- score_f_i (s) = The score of the feature i

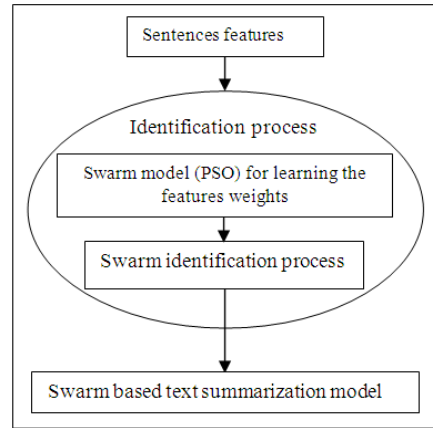


Fig. 2: Training the model

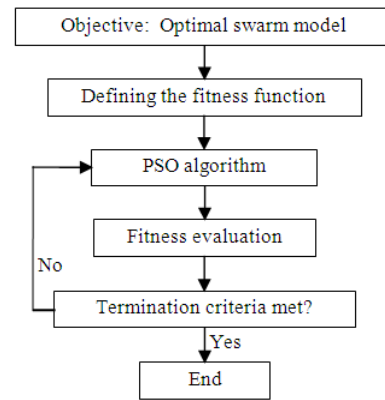


Fig. 3: Swarm identification process

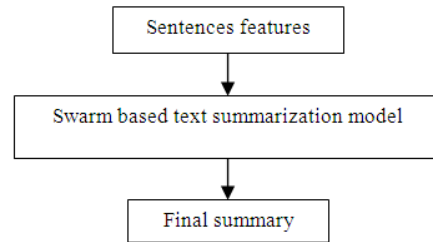


Fig. 4: Testing the model

For more details and the experimental results refer to^[25]. In this study, we employ the fuzzy logic for scoring the sentences instead of the formula in (6).

The fuzzy swarm based summarization: To implement our fuzzy system, we use built-in Mamdani’s fuzzy inference method^[22] of Matlab fuzzy logic toolbox. Below the main parts of the fuzzy inference process are described:

Fuzzification: The inputs are crisp numerical values of five features used in our, those values are limited to the universe of discourse in the range [0, 1]. The features values are adjusted using the weights resulting in the training of the particle swarm optimization (PSO; this forms the central point of merging of the fuzzy logic with swarm intelligence. To determine the degree to which the input values belong to each of the appropriate fuzzy sets, we use the trapezoidal membership function due to its simplicity and widely use. Three fuzzy sets are used: Low, medium and high.

The trapezoidal membership function contains four parameters (a, b, c and d) with the four breakpoints of the trapezium which determine the shape of the function. Moreover the membership function is described by the two indices i and j. For example, the membership function $A_{ij}(a_{ij}, b_{ij}, c_{ij}, d_{ij})$ belongs to the i^{th} fuzzy set and the j^{th} input variable. $B_i(a_i, b_i, c_i, d_i)$ is the output membership function of the i^{th} fuzzy set. The trapezoidal curve is a function of a vector, x , (the j^{th} fuzzy variable) in the i^{th} fuzzy set and depends on the four scalar parameters a, b, c and d, as given by:

$$A_{ij}(x_j) = \begin{cases} \frac{x_j - a_{ij}}{b_{ij} - a_{ij}}, & \text{if } a_{ij} < x_j < b_{ij} \\ 1, & \text{if } b_{ij} \leq x_j < c_{ij} \\ \frac{d_{ij} - x_j}{d_{ij} - c_{ij}}, & \text{if } c_{ij} \leq x_j < d_{ij} \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

where, $a_{ij} \leq b_{ij} \leq c_{ij} \leq d_{ij}$ must hold.

Or in short form:

$$A_{ij}(x_j; a_{ij}, b_{ij}, c_{ij}, d_{ij}) = \max \left(\min \left(\frac{x_j - a_{ij}}{b_{ij} - a_{ij}}, 1, \frac{d_{ij} - x_j}{d_{ij} - c_{ij}} \right), 0 \right) \quad (8)$$

The parameters a and d locate the “feet” of the trapezoid and the parameters b and c locate the “shoulders.”

The output of the trapezoidal membership function is a fuzzy degree of membership (in the range [0, 1]) in the fuzzy set. Figure 5 shows the membership functions of fuzzification of the input value of the Sentence Centrality feature (SC).

Inference: The facts resulted in the fuzzification step need to be merged with a series of the production rules (IF-THEN rules) to perform the fuzzy reasoning process; we defined around 200 IF-THEN rules for that purpose. The following is an example for those rules:

- If (WSS is H) and (SC is H) and (S_FD is M) and (SS_NG is H) and (KWRD is H) then (output is important)

The antecedent of the fuzzy rule in this example has more than one part. To get the output of such antecedent rule, the fuzzy operator is applied to obtain one number which will then be applied to the output function. We use AND operator, it was set as min (minimum) to select the antecedent part with minimum value as output of the antecedent rule. The output of the antecedent rule is used as input for implication process. In this process, we use min (minimum) to reshape the output fuzzy set by truncating it; the output fuzzy membership function is used in this study is the trapezoid membership function as shown in the Fig. 6.

The implication process is implemented for each fuzzy rule. The next sub-step in the inference process is aggregation of outputs of all fuzzy rules and combining them into a single fuzzy set which represents the final output variable.

Defuzzification: The last step in the fuzzy inference process is the defuzzification which is to convert the fuzzy results of the inference into a crisp output which represents the final score of the sentence.

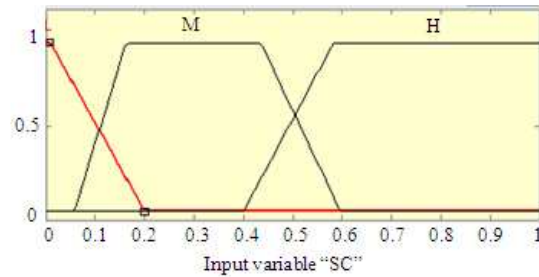


Fig. 5: The trapezoid membership functions of the Sentence Centrality feature (SC)

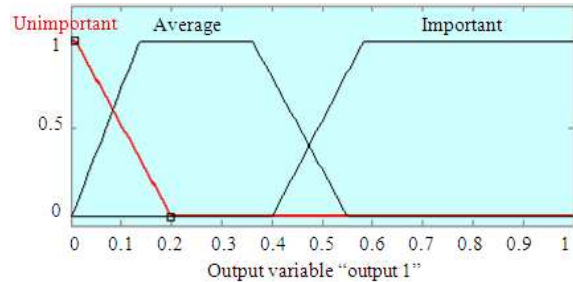


Fig. 6: The trapezoid membership function of the output

We use the centroid method^[26] for defuzzification Eq. 9, which returns the center (one crisp number) of the area under the curve of the output fuzzy set resulting in the aggregation process:

$$z = \frac{\sum_{j=1}^q Z_j u_c(Z_j)}{\sum_{j=1}^q u_c(Z_j)} \quad (9)$$

Where:

z = The center of mass

u_c = The membership in class c at value z_j

After getting the scores of all sentences produced by the fuzzy inferences system, the sentences are reranked based on those scores in descending order, then the top n sentences are selected as summary, where n is equal to the compression rate which is 20% of the total number of the document sentences.

Generalizing the proposed method results via confidence limits: The aim of generalization is to get one value which can express all values in the population of the results. For each summary, evaluation values (recall, precision and f-measure) are created using the evaluation measure ROUGE^[27]. Measuring the performance of the proposed method needs to check each evaluation value separately. Doing so is tough job and a waste of resources. The solution is to use the sample of results (summaries evaluation values) to calculate a range within which any value in the population is likely (95% of the time) to fall. Therefore the range is called the 95% confidence interval. The minimum and maximum values in that range are called the confidence limits. The interval is all values between the confidence limits. The ROUGE^[27] generalizes the evaluation results using bootstrapping (resampling) method.

RESULTS

We evaluate the proposed method using the DUC 2002 document sets (D061j, D062j, D063j, D064j, D065j, D066j, D067f, D068f, D069f, D070f, D071f, D072f, D073b and D077b)^[24] comprising 100 documents.

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) toolkit^[27] is used for evaluation, where ROUGE compares a system generated summary against a human generated summary to measure the quality. ROUGE is the main metric in the DUC text summarization evaluations. It has different variants, in our experiment, we use ROUGE-N (N = 1 and 2) and

ROUGE-L, the reason for selecting these measures is what was reported by same study^[27] that those measures work well for single document summarization.

In DUC 2002 document sets, each document set contains two model or human generated summaries for each document. We gave the names H1 and H2 for those two model summaries. The human summary H2 is used as benchmark to measure the quality of our proposed method summary, while the human summary H1 is used as reference summary. Beside the human with human benchmark (H2-H1) (H2 against H1); we also use another benchmark which is MS word summarizer (Msword).

Table 1 shows a comparison between the proposed method evaluation and the other three methods (the swarm model and the two benchmarks (Msword and H2-H1)) based on the average recall, precision and F-measure using ROUGE-1, ROUGE-2 and ROUGE-L, where those averages for the four methods (the proposed method, the swarm model, Msword and H2-H1) were generalized using the confidence limits (95%-confidence interval). The Fig. 7-9 visualize the same results drawn in the Table 1.

Table 1: The fuzzy swarm, swarm model, ms word summarizer and H2-H1 comparison: average recall using ROUGE-(1, 2 and l) at the 95%-confidence interval

ROUGE	Method	Avg-R	Avg-P	Avg-F
1	Msword	0.39306	0.48487	0.42477
	Swarm model	0.43028	0.47741	0.44669
	Fuzzy swarm	0.43622	0.49126	0.45524
	H2-H1	0.49657	0.49613	0.49605
2	Msword	0.16325	0.21066	0.17947
	Swarm model	0.18828	0.21622	0.19776
	Fuzzy swarm	0.19702	0.23037	0.20847
	H2-H1	0.20957	0.20940	0.20938
L	Msword	0.36605	0.45272	0.39604
	Swarm model	0.39674	0.44143	0.41221
	Fuzzy swarm	0.40144	0.45355	0.41937
	H2-H1	0.46524	0.46490	0.46479

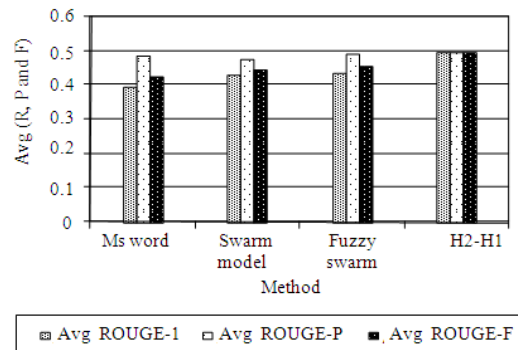


Fig. 7: The fuzzy swarm, swarm model, ms word summarizer and H2-H1 comparison: Average recall using ROUGE-1

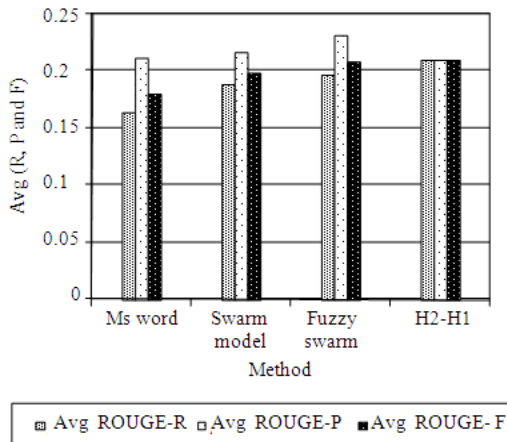


Fig. 8: The fuzzy swarm, swarm model, Msword summarizer and H2-H1 comparison: Average recall using ROUGE-2

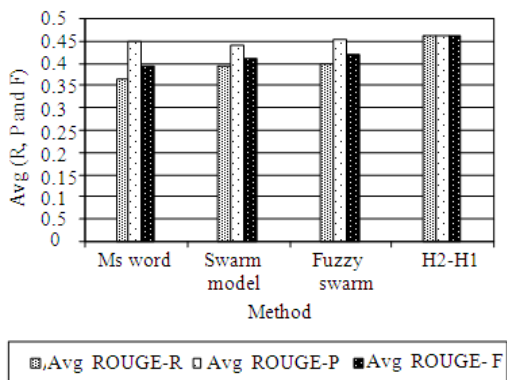


Fig. 9: The fuzzy swarm, swarm model, Msword summarizer and H2-H1 comparison: Average recall using ROUGE-L

The purpose of using the human summarizer (H2-H1) as benchmark is to show how much the performance of the proposed method, the swarm model and Msword summarizers is acceptable compared with that performance of the human (H2-H1).

Based on the generalization of the results obtained by the four methods (the proposed method, the swarm model, Msword and H2-H1), the proposed method performs better than the swarm model and the Msword summarizer. The low overlapping between the summaries generated manually by the human makes achieving high evaluation values difficult even for human summarizer; we found that the overlapping between the two human summaries which we used in this study is 49% similar to each other.

DISCUSSION

The experimental results lead to two interesting observations. Firstly, the proposed method based on fuzzy logic and swarm intelligence (PSO) for text summarization problem showed good performance compared to other methods used in this study. Secondly, the low overlap between the summaries generated manually by the humans made achieving high evaluation values difficult. For instance, we found that the overlapping between the two human summaries (H2 and H1) which we used in this study is 49% similar to each other. The weights suggested by PSO promoted the scores of the highly important features, which give each text feature the right score it was worth. The experimental results supported the incorporation of fuzzy logic with swarm intelligence to make the risks, uncertainty, ambiguity and imprecise values for choosing the weights (scores) of the text features to be flexibly tolerated. For our future study, we will incorporate the proposed method with diversity based methods in a different hybrid model.

CONCLUSION

In this study, we introduced a method based on fuzzy logic and swarm intelligence (PSO) for text summarization problem. The weights suggested by PSO were used to adjust the text features scores. The fuzzy inference system was employed to use the adjusted features scores as inputs, based on which the sentences are evaluated and the most relevant sentences are selected to be included in the summary. The results showed that the proposed method has better performance outperforming the swarm model and the benchmark methods used in this study.

ACKNOWLEDGMENT

This project is sponsored partly by the Ministry of Science, Technology and Innovation under E-Science grant 01-01-06-SF0502, Malaysia.

REFERENCES

1. Luhn, H.P., 1958. The automatic creation of literature abstracts. *IBM J. Res. Develop.*, 2: 159-165. <http://domino.research.ibm.com/tchjr/journalindex.nsf/c469af92ea9eaceac85256bd50048567c/97e0420635a5000a85256bfa00683d33?OpenDocument>
2. Kupiec, J., J. Pedersen and F. Chen, 1995. A trainable document summarizer. *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, July 9-13, New York, USA., pp: 68-73. <http://portal.acm.org/citation.cfm?id=215333>

3. Lin, C.Y. and E. Hovy, 1997. Identifying topics by position. Proceedings of the 5th Conference on Applied Natural Language Processing, Mar. 31-Apr. 03, San Francisco, CA., USA., pp: 283-290. <http://portal.acm.org/citation.cfm?id=974557.974599>
4. Lin, C.Y., 1999. Training a selection function for extraction. Proceedings of the 18th Annual International ACM Conference on Information and Knowledge Management Nov. 2-6, Kansas City, Kansas, pp: 55-62. <http://portal.acm.org/citation.cfm?id=319950.319957>
5. Conroy, J.M. and D.P. O'leary, 2001. Text summarization via hidden markov models. Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Sept. 9-12, New Orleans, Louisiana, United States, pp: 406-407. <http://portal.acm.org/citation.cfm?id=383952.384042>
6. Osborne, M., 2002. Using maximum entropy for sentence extraction. Proceedings of the ACL'02 Workshop on Automatic Summarization. July 11-12, Morristown, New Jersey, USA., pp: 1-8. <http://portal.acm.org/citation.cfm?id=1118163>
7. Svore, K., L. Vanderwende and C. Burges, 2007. Enhancing single-document summarization by combining RankNet and third-party sources. Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, June 2007, Association for Computational Linguistics, Prague, pp: 448-457. <http://research.microsoft.com/en-us/um/people/cburges/papers/emnlp07.pdf>
8. Fattah, M.A. and F. Ren, 2009. GA, MR, FFNN, PNN and GMM based models for automatic text summarization. *Comput. Speech Language*, 23: 126-144. <http://portal.acm.org/citation.cfm?id=1393839>
9. Binwahlan, M.S., N. Salim and L. Suanmali, 2009. Swarm based features selection for text summarization. *Int. J. Comput. Sci. Network Secur.*, 9: 175-179. http://paper.ijcsns.org/07_book/200901/20090125.pdf
10. Kennedy, J. and R. Eberhart, 1995. Particle swarm optimization. Proceedings of the IEEE International Conference on Neural Networks, Nov. 27-Dec. 1, IEEE Xplore Press, Perth, Australia, pp: 1942-1948. DOI: 10.1109/ICNN.1995.488968
11. Ziegler, C. and M. Skubacz, 2007. Content extraction from news pages using particle swarm optimization on linguistic and structural features. Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, Nov. 2-5, IEEE Computer Society Washington, DC., USA., pp: 242-249. <http://portal.acm.org/citation.cfm?id=1331850>
12. Merwe, V.D. and A.P. Engelbrecht, 2003. Data clustering using particle swarm optimization. Proceedings of the IEEE Congress on Evolutionary Computation, Dec. 8-12, IEEE Xplore Press, Canbella, Australia, pp: 215-220. DOI: 10.1109/CEC.2003.1299577
13. Cui, X., T.E. Potok and P. Palathingal, 2005. Document clustering using particle swarm optimization. Proceedings of the IEEE Swarm Intelligence Symposium, June 8-10, IEEE Xplore Press, Pasadena, California, pp: 185-191. DOI: 10.1109/SIS.2005.1501621
14. Wang, Z., Q. Zhang and D. Zhang, 2007. A pso-based web document classification algorithm. Proceedings of the IEEE 8th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, July 30-Aug. 1, IEEE Xplore Press, Qingdao, China, pp: 659-664. DOI: 10.1109/SNPD.2007.72
15. Kiani, B.A. and T.M.R. Akbarzadeh, 2006. Automatic text summarization using: Hybrid Fuzzy GA-GP. Proceedings of the IEEE International Conference on Fuzzy Systems, July 16-21, IEEE Xplore Press, Vancouver, BC., Canada, pp: 977-983. DOI: 10.1109/FUZZY.2006.1681829
16. Kyoomarsi, F., Khosravi, H., Eslami, E., Dehkordy, P., K. and Tajoddin, A. (2008). Optimizing Text Summarization Based on Fuzzy Logic. Proceedings of the Seventh IEEE/ACIS international Conference on Computer and information Science. 14 - 16 May. Washington, DC, USA: IEEE Computer Society, 347-352. DOI: 10.1109/ICIS.2008.46
17. Eberhart, R.C. and J. Kennedy, 1995. A new optimizer using particle swarm theory. Proceedings of the 6th International Symposium on Micro Machine and Human Science, IEEE Service Center, Piscataway, NJ., Nagoya, Japan, pp: 39-43. <http://www.citeulike.org/user/wtribbey/article/1439842>
18. Eberhart, R.C. and Y. Shi, 2001. Particle swarm optimization: Developments, applications and resources. Proceedings of the 2001 Congress on Evolutionary Computation, May 27-30, Seoul, IEEE Xplore Press, Korea, pp: 81-86. DOI: 10.1109/CEC.2001.934374
19. Kennedy, J. and R.C. Eberhart, 1997. A discrete binary version of the particle swarm algorithm. Proceedings of the IEEE International Conference on Computational Cybernetics and Simulation, (ICCCS '97), IEEE Xplore Press, New York, pp: 4104-4108. <http://www.citeulike.org/user/peskin/article/2299544>

20. Zadeh, L.A., 1965. Fuzzy sets. *Information and control.*, 8: 338-353.
<http://www-bisc.cs.berkeley.edu/Zadeh-1965.pdf>
21. Klir, G.J. and B. Yuan, 1995. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. 1st Edn., Prentice-Hall, New York, ISBN: 10: 0131011715, pp: 592.
22. Mamdani, E.H. and S. Assilian, 1975. An experiment in linguistic synthesis with a fuzzy logic controller. *Int. J. Man. Mach. Stud.*, 7: 1-15.
http://www.eric.ed.gov/ERICWebPortal/custom/portlets/recordDetails/detailmini.jsp?_nfpb=true&_ERICExtSearch_SearchValue_0=EJ114552&ERICExtSearch_SearchType_0=no&accno=EJ114552
23. Sugeno, M. and T. Takagi, 1985. Fuzzy identification of systems and its application to modeling and control. *IEEE. Trans. Syst. Man. Cybernet.*, 15: 116-132.
<http://www.citeulike.org/user/cmowes/article/3928474>
24. NIST, 2002. The Document Understanding Conference (DUC).
http://www.linguatca.pt/aval_conjunta/Faro2002/HTML/Alexsandro_Soares/sld014.htm
25. Binwahlan, M.S., N. Salim and L. Suanmali, 2009. Swarm based text summarization. *Proceedings of the International Conference on IACSIT Spring Conference*, Apr. 17-20. Singapore, pp: 145-150. DOI: 10.1109/IACSIT-SC.2009.61
26. Sivanandam, S.N., S. Sumathi and S.N. Deepa, 2006. *Introduction to Fuzzy Logic using MATLAB*. 1st Edn., Springer-Verlag, New York, ISBN: 10: 3540357807, pp: 430.
27. Lin, C., 2004. Rouge: A package for automatic evaluation of summaries. *Proceedings of the Workshop on Text Summarization Branches Out, 42nd Annual Meeting of the Association for Computational Linguistics*, July 25-26, Barcelona, Spain, pp: 74-81.
<http://www.aclweb.org/anthology-new/W/W04/W04-1013.bib>