The Similarities Test of Thesis by Percentage of Similarity Formulation

Poltak Sihombing

***Abstract*— In this paper, a search engine system based Genetic Algorithm (GA) which used to find and rank the similarity percentage of some thesis titles in the database is developed. In the simulation, the sources of the thesis titles were obtained from Faculty of computer science and information technology, University of Sumatera Utara Medan and the paper collection of "Badan tenaga atom nasional" Jakarta. A keyword solution population whis is produced by GA was used to calculate the percentage of similarity (POSI) of the thesis titles. Furthermore the POSI also used to rank the thesis titles based on the percentage of similarity obtained in the previous process. The simulation result shows that the POSI accuracy about 92% was obtained.**

***Index Terms*— POSI formulation, GA scheme, keyword competition, similarity**

I. INTRODUCTION

IN a rank of retrieval thesis, a designed computer system managing a thesis collection which contains text and others possibly media. The collections are typically quite large size (megabytes or gigabytes) of text and consists of a thousands or more number of the thesis. A retrieval system is nolmally designed to cover a number of things such as displaying a thesis to a user, managing the database of thesis, and also provide a browsing capabilities. The most important in a retrieval system is due to its ability to accuratly provide the required information of a thesis to a user in a short of time and to estimate the degree of relevance of each thesis to the user's query. The estimated degree of relevance are then used to rank the thesis for the user from the most to the less similar. One of the weaknesses of the current Information Retrieval System (IRS) is that the system has not had the ability to provide a percentage of the value of the similarity thesis drawn from the database, so users tend to be difficult to distinguish the thesis that the most relevant (similar). To solve this problem, some researchers have applied several methods such as Boolean model, vector space model, a probabilistic model, fuzzy retrieval models, and models based on artificial intelligence techniques. [1- 3].

In this paper, two methods namely GA and POSI which used to generate keyword solution through query competition and to sequence the similarity percentage of the thesis entitled as a query against the thesis title in the database, respectively, is proposed. The produced keywords solutions by the GA were calculated and formulated by POSI on every document that exists in the database [4, 5]. The reasons of using the GA method is due to its ability to optimize the retrieval thesis using keyword competition [6, 7]. In the the GA processing, the keywords of the thesis is then converted into a bit string known as genes. Furthermore, a collection of several genes called chromosomes by mean the gene is part of a chromosome [8-10].

Manuscript received July 10, 2015. This paper with title "The Similarities Test of Thesis by Percentage of Similarity Formulation" was supported in part by the Decentralized Research Program, DIKTI, Indonesia, 2015.

Poltak Sihombing is with the Faculty of Computer Science and Information Technology, University of Sumatera Utara, Medan, Indonesia (phone:+62618210077; fax: +62618210077; mobile phone: 081260889005, e-mail: poltak@usu.ac.id, poltakhombing@yahoo.com).

II. METHODOLOGY

A. The Methodology of the System

Fig. 1 shows the methodology scheme of the developed system. Through keyword competition, the keywords solution (KS) are produced by the GA method from the input query (the thesis title). By using the obtained keyword solution (with GA) of each thesis in the database, its frequency appearance is counted (with POSI formulation). Furthermore, the posi method used to sort the retrieval thesis based on the similarity percentage of each thesis in the database against the thesis that exist within the query input. The developed IRS system can be used to see the similarity percentage of the entered thesis title into the system against the stored thesis title in database.

B. Keyword Competition in GA Process

An individuals population of the keyword competition (KC), $P(t) = x_1, \ldots, x_n$ at iteration t, is maintains of the GA processes. Each individual which is implemented as data structure S, is used to represents a potential solution to the faced problem. Each solution x_i is evaluated to see the level fitness. With crossover and mutation technigues, then a new population at iteration t+1 is produced by selecting the stronger individuals (the higher fitness). The performance produced population is hightly influenced by the crossover and mutation operators [11].

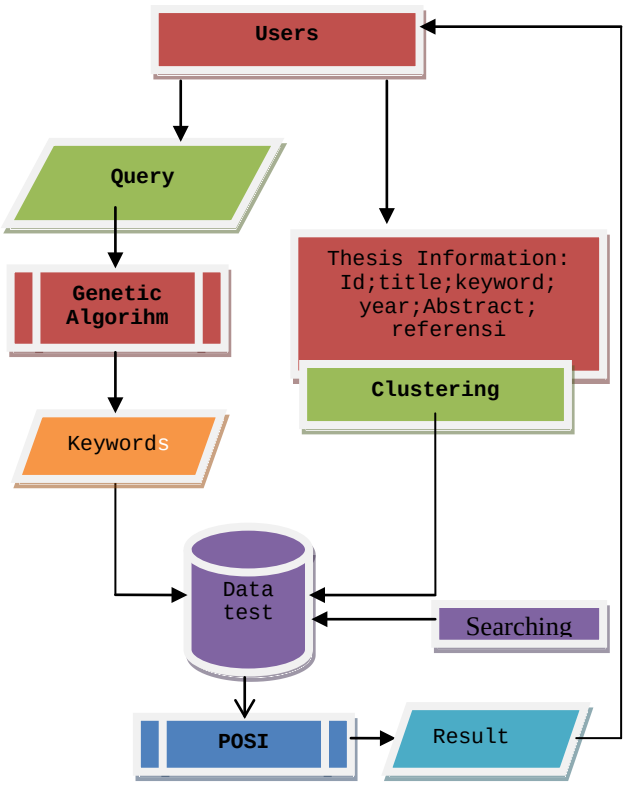

Fig.1. The methodology scheme of the developed system

In general, a lot of keywords is given as a query. The KC is used to remove the keywords that are less essential and leave only the keywords that are considered most important (keywords that are most influential in accessing the relevant documents in the database). The developed GA system is used to compete all the selected keywords each other through the process of chromosome formulation, population initialization, fitness evaluation, parents selection, crossover (married parents), and mutation (individual mutate). The flow of the keyword kompetition in the GA system is shown in Fig. 2.

C. *Encoding of a Chromosome*

Given n queries (n is an integer) where each query Q_i consist of j keywords and represented by q_{ij} (the j^{th} keyword of document i,where i and j are an integer 1,2,3…, n). In this paper, the chromosome is represented by keyword as: Q_0, Q_1, Q_2, ...,Q_n, and hence the queries are as

$$\begin{aligned} Q_0 &= (q_{01}, q_{02}, q_{03}, ..., q_{0j}) \\ Q_1 &= (q_{11}, q_{12}, q_{13}, ..., q_{1j}) \\ Q_2 &= (q_{21}, q_{22}, q_{23}, ..., q_{2j}) \\ Q_n &= (q_{i1}, q_{i2}, q_{i3}, ..., q_{ij}) \end{aligned} \quad (1)$$

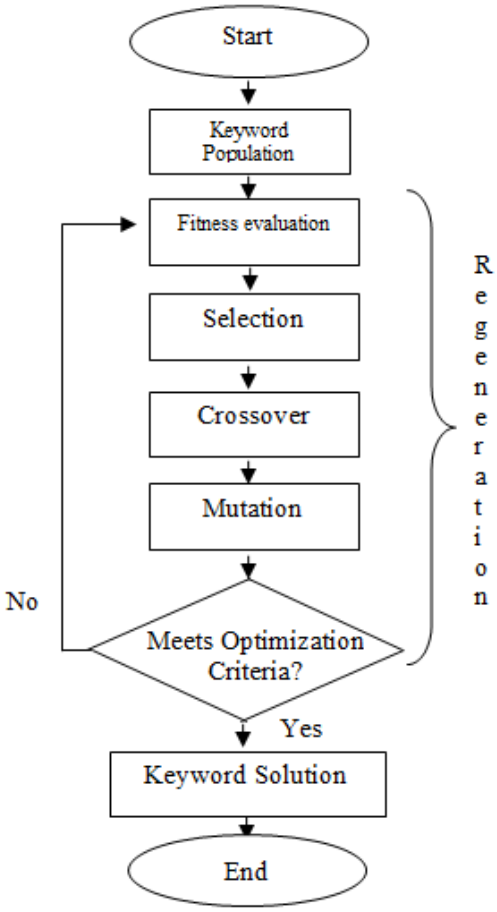

Fig. 2 Keyword Competition in the GA Process

Therefore the chromosome length can be writtens as $Q_{\text{total}} = \{Q_0 \cup Q_1 \cup Q_2 \cup .... Q_n\}$. Lets take an example of five papers called paper-0, paper-1, paper-2, paper-3, and paper-4. Then, they are represented by DOC-0, DOC-1, DOC-2, DOC-3, and DOC-4, respectively.

D. *Percentage of Similarity Formulation*

The author proposed the similarity percentage called percentage of similarity (POSI) formulation. In the proposed method, let assume that the appearance of KS (k_1) in thesis 1 be k_1 x d_1, the appearance of KS (k_2) in thesis 2 be k_2 x d_2, the appearance of KS (k_i) in thesis j be k_i x d_j, and the appearance of total KS ($k_{i,...n}$) in all of the thesis collection be K_{total}. Therefore, the POSI formulation iswriten in the following equation

$$\text{Sim}(k,d) = \frac{\sum_1^n k_i d_j}{K_{\text{total}}}, \quad (2)$$

where Sim (k,d) indicates the percentage of similarity value, $k_i d_j$ indicates the sum of each KS which found in the thesis collection, and K_{total} indicates the appearance total of all keywords solution which found in all of the title, abstract, and keyword of thesis collection.

In the POSI formulation (equation 2), it can be seen that KS of query is matched to the text of thesis collection in database. The next matching process is as follow; let the list of keyword solution (query) represented in the binary as 0111010100 and called as keywords k_n (n = 1, ..., 9). For example, the KS: (0,1,1,1,0,1,0,1,0,0), it's mean (k_0=0. k_1=senyawa, k_2=pirofosfat, k_3=merah, k_4=0, k_5=teknesium, k_6=0, k_7=jantung, k_8=0, k_9=0). Then their matching process are k_1, k_2, k_3, k_5, and k_7. If the keyword (k) is 0, then it will automatically excluded from the KS. The system then will match those keywords solution by finding the appearance of those keywords in each thesis collection in the database. For example, the keyword solution 'senyawa' (k_1) will match to the word (term) of 'senyawa' in the thesis collection. Continuously, the system wil searches the same word to the next other keyword solution and counts them with the POSI formulation in the equation 2.

III. THE PROTOTYPE OF POSI FORMULATION

The prototype of the POSI formulation is developed. In the prototype, the POSI formulation is used determine the similarity measurement of the retrieved thesis from the database. The developed POSI prototype system is shown in Fig. 3.

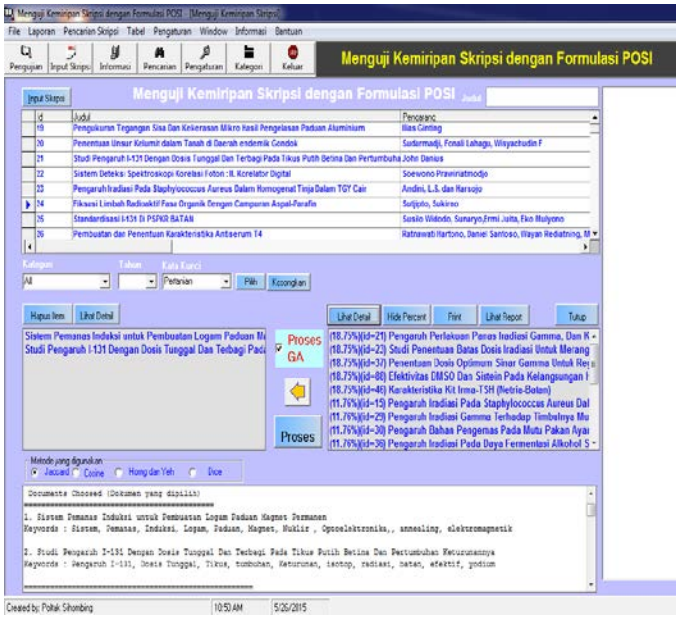

Fig.3. The prototype of the POSI formulation

A. Thesis Ranking

The similarity percentage formulation of the entered thesis by calculating the appearance of the KS (k) in each thesis collection (d) in database is genereated. The appearance of KS (k_i) in the thesis j is represented by $k_i d_j$ and the total appearance of KS ($k_{i,...n}$) in all of the thesis collection is represetned by K_{total}. Further, the rank of the thesis is sorted based on calculation results of the similarity percentage. The top position of the retrieved thesis is started from the highest value of the similarity percentage.

B. The similarity calculation

Let's take an example of user's queries, keyword process, keyword solution, and retrieved thesis as shown in Fig. 4. The result shows that there are two thesis used as a user's query which each thesis is represented by a set of keywords. In this step, the user want to retrieve the thesis that predicted similar with the user's query in the database. It can be seen that the set of all keywords (user's query) in Fig. 4 are as follows: Kriptografi, Asimetris, Algoritma ElGamal, Elias Gamma Code, Fermat, Reduksi Noise, Kombinasi Harmonic Mean Filter, Contraharmonic Mean Filter, Gaussian Noise, Rayleigh Noise, Spackle Noise, and Uniform Noise.

```
USER QUERY:
==========
1. Text Data Encryption with El Gamal Algorithm and
Data Compression Algorithm by Elias Gamma Code
Algorithm.
Keywords: Kriptografi, Asimetris, Algoritma
ElGamal, Elias Gamma Code, Fermat
2. Implementation of Combining Harmonic Mean Filter
Method and Mean Contraharmonic Filter to Reduce
Noise on Digital Image.
Keywords : Reduksi Noise, Kombinasi Harmonic
Mean Filter, Contraharmonic Mean Filter,
Gaussian Noise, Rayleigh Noise, Spackle Noise,
Uniform Noise.
==============================
Set of all Keywords as a query:
==============================
Kriptografi, Asimetris, Algoritma ElGamal,
Elias Gamma Code, Fermat, Reduksi Noise,
Kombinasi Harmonic Mean Filter, Contraharmonic
Mean Filter, Gaussian Noise, Rayleigh Noise,
Spackle Noise, Uniform Noise.
```

Fig. 4. The user's queries in the GA process

In the next of the GA processing, the first and the second generation population of the keyword competition approach are generated as shown in Fig. 5 and Fig. 6, respectively.

```
1st Generation
===============
First Population:
11111111000000000 [0, 529411764705882]
00000001111111111 [0, 529411764705882]
Population after selection:
00000001111111111
11111111000000000

Crossover Process:
Chromosome 2 with chromosome 1, position 16
Population after crossover:
00000001111111110
11111111000000001

Mutation Process:
Chromosome 1 position 5
Population after mutation:
00001001111111110
11111111000000001
```

Fig.5. The keyword competition in the GA process

```
2nd Generation
==============
First Population:
00001001111111110 [0, 471590909090909]
11111111000000001 [0, 503267973856209]
Population after selection:
11111111000000001
00001001111111110

Population after crossover:
11111111000000001
00001001111111110

Mutation Process:
Chromosome 1 position 3
Population after mutation:
11011111000000001
00001001111111110
```

Fig.6. The second generation in the GA process

From Fig. 6, the first populations is represented in the second generation as follows:

00001001111111110
11111111000000001

and the populations after selection are as follows:

11111111000000001
00001001111111110

In the GA process showed that population after selection is same with population after crossover. It is mean that there is no crossover in the second generation. Therefore there is no change of chromosome; the populations are still the same with the chromosome after selection as

11111111000000001
00001001111111110

In the next process, it can be seen that the mutation is happened in the first chromosome with bit position about 3. Therefore the populations after mutation are as follows:

11011111000000001
00001001111111110

This population will be generated as early population which called as a first population in the third generation as shown in the Fig.7. The next process of the GA is shown in the Fig. 8. The result chromosome of the keyword solution is written as 11011111000000001. Then, the list of the keyword solution resulted by GA Process are listed as Kriptografi, Algoritma ElGamal, Fermat, Gaussian Noise, Rayleigh Noise, Spackle Noise, and Uniform Noise. In the next step, the keyword competition process has reached the stable chromosome and the last generation is obtained as 11011111000000001. The last generation is given in the Fig. 8.

```
 3rd Generation
==============
First Population:
11011111000000001 [0, 451388888888889]
00001001111111110 [0, 471590909090909]

Population after selection:
11011111000000001
11011111000000001
Crossover Process:
Chromosome 2 with chromosome 1, position 2
Population after crossover:
11011111000000001
11011111000000001
Mutation Process:
Population after mutation:
11011111000000001
11011111000000001
```

Fig.7. The third generation in the GA process

```
LAST GENERATION
===============
Chromosome of Keyword Solution:
11011111000000001
Keywords Solution:
Kriptografi, Algoritma ElGamal, Fermat,
Gaussian Noise, Rayleigh Noise, Spackle Noise,
Uniform Noise.

Process Time  : 1 second
```

Fig.8. The keyword solution (KS) resullted by the GA Process

IV. THE POSI RESULTS

In the last process of the GA, the Keyword Solution such as Kriptografi, Algoritma ElGamal, Fermat, Gaussian Noise, Rayleigh Noise, Spackle Noise, and Uniform Noise were obtained. Those KS will be linked to the thesis collection in the database. In the next process, the system will count how many times the appearance of the KS and rank the retrieved thesis according to each KS as shown in the POSI formulation (equation 2). The system then present the retrieved thesis as shown in Table I.

TABLE I
THE POSI RESULT

Rank of thesis	Similarity (%)	Thesis- Id	The Title of thesis Retrieved
1	31.17	1565	El Gamal algorithm for Security SMS On Android
2	23.51	1572	Comparative Analysis of Geometric Mean Filter with Sobel operator, Prewitt Operator and the Operator Robert in Bitmap image.
3	15.67	1553	Midpoint combination of Gaussian Smoothing Filters and Filter to

			Improve the Quality of Digital Image
4	10.23	1567	Implementation of k-Nearest Neighbor algorithm to classify Batik Besurek from Bengkulu
5	8.03	1550	Comparison of Canny Methods, Robert and Laplacian of Gaussian On Camera image results for 360

Table I presents the example of the similarity calculation of the retrieved thesis from the database. It was find out that the similarity percentage were about 31.17 %, 23.51%, 15.67%, 10.23 %, 8.03% for the thesis-Id 1565**,** 1572, 1553, 1567, and 1550, respectively. The sorted list from the highest into the lower similarity percentage were automatically obtained by the proposed POSI formulation.

V. EXPERT EVALUATION

The percentage of similarity of documents retrieved in Table I has been evaluated by experts as represented in The Table II. The documents with id-doc number 1565, 1572, 1553 are the most potential similar to those keywords. Therefore, we believe that other documents (id-doc number 1560 and 1550) are predicted similar. The expert evaluation is presented in the three categories called S indicates Similar, PS indicates Predicted Similar and SM indicates Similar Marginal.

TABLE II
EXPERT EVALUATIONS

Rank of Doc	Similarity (%)	Id-Doc	Expert evaluation
1	31.17	1565	S
2	23.51	1572	S
3	15.67	1553	S
4	10.23	1567	PS
5	8.03	1550	SM

According to the expert evaluation in Table II, it can be seen that the id-doc 1565, 1572, and 1553 are the most similar to the query. Those results is enough to prove that the keyword solution (KS) maight be used to get the most potential similar document from the document collections in the database as proposed in this paper. Several queries have been tested with different formulations. The result shows that similarities level of the retrieval documents are consistent even though the percentage of the similarity could be change.

The simulation results described that the used the keyword competition approach and keyword based ranking scheme by POSI formulation could accurately identify the most similar document to user's query. The ranking of the percentage similarity in each retrieval documents will facilitate the users to select their required document. It is also found that if the sum of generation is increased then the process time also will be increased. Furthermore, it is observed that if the sum of query increased, therefore the sum of generation, crossover, mutation, and process time also increased. However, if the sum of queries, crossover, and mutation are increased then the similarity percentage of the retrieval document is not increased.

VI. CONCLUSION AND FUTURE WORK

A. Conclusion

From the research result it can be seen obviously the developed system robustly work in retrieved the required thesis from the database. Beside provide the similarity percentage of the entered input query to the thesis title resided in the database, also it could automatically rank the similarity of the thesis title saved in the database. Hopefully this system can be developed further and used by public to prevent and stop plagiarism.

B. Future work

The developed prototype is expected to be improved further into a large scope system such as multi-dimensional access to the information within the network and build digital libraries with multi-search in larger databases. It can be also extend to the broad application such as plagiarism test.

ACKNOWLEDGMENT

The authors would like to thank the Head of Department and staff of "Direktorat Penelitian dan Pengabdian kepada Masyarakat (DP2M) Dirjen Pendidikan Tinggi (DIKTI)", Jakarta for the excellent services to support this research. To the Chief of the University of Sumatera Utara Research Centre; Prof. Dr. Ir. Harmein Nasution, MSIE. To the Dean and staff of Faculty of Computer Science and Information Technology; University of Sumatra Utara, Prof. Dr. Muhammad Zarlis, MSc for their widest to support this work.

REFERENCES

[1] V. Hatzivassiloglou, L. Gravano, and A. Maganti, "An investigation of linguistic features and clustering algorithms for topical thesis clustering," *Proceedings of the 23rd annual ACM SIGIR conference*, Athens, Greece, 2000, , pp. 224-231.

[2] P. Sihombing, A. Embong, and P. Sumari, "A technique of probability in thesis similarity comparison in IRS," *Proceedings of the IMT-GT (Indonesia Malaysia Thailand) conference*, Parapat, Indonesia. June 13-14, 2005.

[3] I. King and K. C. Sia, "Distributed content-based visual information retrieval system on peer-to-peer networks," *Journal of ACM transactions on information systems*, vol. 22, no. 3, 2004, pp. 477–501.

[4] P. Sihombing, P. Sumari, and A. Embong, "A comparison of thesis similarity in information retrieval system by different formulation," *Proceedings of the IMT-GT (Indonesia Malaysia Thailand) conference*, Penang, Malaysia. June, 2006.

[5] M. Carey, F. Kriwaczek, S. M. Rüger, "A visualization interface for thesis searching and browsing," *Proceedings of the ACM CIKM 2000 workshop on new paradigms in information visualization and manipulation*, Washington, DC, 2000.

[6] J. R. Koza, "*Genetic Programming: On the programming of computers by means of natural selection"*. Cambridge, MA, USA: The MIT Press, 2005.

[7] K. S. Jones, "The role of artificial intelligence in information retrieval," *Journal of the American society for information science*, vol. 42, no. 8, 2001, pp.558-565.
[8] D. E. Goldberg, "*Genetic Algorithms in search, optimization, and machine learning,"* Addison-Wesley, Reading, MA, 2005.
[9] Y. Kural, Robertson, S. Jones, "Deciphering cluster representations," *Information processing & management*, vol. 37, no. 4, 2001, pp. 593-60.
[10] M. Gordon, "Probabilistic and genetic algorithms in thesis retrieval," *Communications of the ACM*, vol. 31, no. 10, 2000.
[11] C. W. Ahn and R. S. Ramakrishna, "A genetic algorithm for shortest path routing problem and the sizing of populations," *IEEE transactions on evolutionary computation*, vol. 6, no. 6, 2002, pp. 566–579.

Poltak Sihombing is a Senior lecturer in Computer Science Department, Faculty of Computer Science and Information Technology, University of Sumatera Utara (USU), Medan, Indonesia. His main research focuses on Artificial Intelligence, Information Retrieval, Software Engineering, Network Computer, and Microcontroller System. He was graduated from Computation Physics, University of Sumatera Utara Medan. He got a master degree from University of Indonesia Jakarta on Computer Science field, and the Ph.D degree in Computer Science from Univeristi Sains Malaysia (USM). He is an active researcher, also member of APTIKOM Indonesia.