Increase Performance Genetic Algorithm In Matching System By Setting GA Parameter

Rijois Iboy Erwin Saragih\(^1\*\) and Darsono Nababan\(^2\)

\(^1\)Universitas Methodist Indonesia, Jl. Hang Tuah No. 8 Medan, 20152, Indonesia
\(^2\)Universitas Pelita Harapan, Jl. Iman Bonjol No. 6 Medan, 20112, Indonesia

Abstract. Matching system is a system that finding an object base on what we are looking for. In this research, the object is a paragraph. Paragraph matching is a process of finding a paragraph which is inputted as a target paragraph. The previous worked on this filed is that focus on finding a sentence. It shows that it took 490 generations or iterations in order to find the sentence, which mean it is not so fast or takes more times. This paper intends to bring into expand research which means from sentence matching into paragraph matching. It means that there are two points to bring about that is paragraph matching and increase performance. On this research used genetic algorithm to get best speed of finding the sentence through several genetic algorithm components that is population, selection, crossover, and mutation. In implementation side, it sets the parameter value of genetic algorithm that is chromosome= 10, 20, 50, crossover probability=0.5 and mutation probability=0.05. The outcome of this research is to give a better approach of finding a paragraph accurately and also faster in term of speed. The final result is that the generation or iterations is 144.

1. Introduction

Paragraph matching is a process of finding a paragraph which is inputted as a target sentence. This area of research is similar to word searching technic. Sentence is built on word and word is built on alphabet. There are many languages in the world. Each language has its own structure both of grammar or pronunciation. In this research English will be used as the target sentence.

Matching, in other word is a process of putting the word as it is required to be a sentence. Word by word is formed until the same with the sentence is typed by the user.

General problem on Genetic Algorithm which often happen is local optima and this problem occur because of population diversity, first population and next population[10]. If the difference of population is too small so it might be occur local optima, and if the difference of population is to big so it causes a long time of genetic algorithm to get best solution. There are many research have done on this field in order to avoid local optima along with increasing performance of Genetic Algorithm. One of the approaches which can be done is through improving the performance of genetic operator such as selection, crossover and mutation.

Genetic algorithms [1] are search and optimization algorithms based on the principles of natural evolution, which were first introduced by john Holland in 1970. Genetic algorithms also implement the optimization strategies by simulating evolution of species through natural selections. Genetic algorithm is generally composed of two processes. First process is selection of individual for the
production of next generation and second process is manipulation of the selected individual to form the next generation by crossover and mutation techniques [2]. The selection mechanism determines which individual are chosen for reproduction and how many offspring each selected individual produce. The main principle of selection strategy is the better is an individual; the higher is its chance of being parent.

Genetic Algorithm is a population class based on random searching technic which used in many practical applications. Usually this algorithm keeps several potential solutions for a problem that is being handled, in which can see as form work memory is recognized as population. Point of new iteration in a space of searching which result for evaluation and optional is inputted in to population [4].

Genetic algorithm performances is influenced by genetic operator; selection operator, crossover and mutation.[5]

The GA has following steps:

1. Initialization: genetic algorithms are [6] generally start with an initial population that is generated randomly some research has been conducted using special technique to produce a higher quality initial population. Thus an approach is designed to give the GA a good start and speed up the evolutionary process.

2. Selection: It select the two parent chromosome from a population according to their fitness better the fitness bigger the chance to be selected.

3. Reproduction: - It selects the two chromosomes according to current selection procedure perform crossover on them and obtain one or two children, perhaps apply mutation as well and install the result back into that population, the least fit of population is destroyed.

4. Crossover: With a crossover probability crossover the parent to form new offspring (children).

5. Mutation:-After a crossover [7] this operator is performed. Mutation is a genetic operator used to maintain genetic diversity from one generation of a population of chromosomes to next.

6. Replacement: Use new generated population for a further run of algorithm.

Genetic Algorithm on forming sentence also called paragraph matching is that genetic searching probability on words appearance. Matching paragraph is done by typing a sentence as an input, and then rising the sentence randomly which is repeatedly till is found the sentence that is inputted on the first time. This matching paragraph is done until it is got the exact value or similar value that is given. The accuracy of this matching is determined by the similarity of target value. On this research genetic algorithm will be implemented to do matching paragraph based on the sentence which is inputted.

Therefore, the researcher is interest to do research on paragraph matching use genetic algorithm.

2. Methodology

On genetic algorithm there is several step of process in order to solve optimization problem s. The steps are: Population, Selection, Crossover and Mutation.

Those steps will continually repeat until finding the result that is wanted. Paragraph matching will form a paragraph using genetic algorithm, it is hoped can give a result that is optimal.

The data on this research is words, and then the genetic algorithm will do forming the words to become a paragraph based on the input.

On population consist of ten chromosomes or populations, this chromosome are ordered by gens which content value. Gen value is determined based on gen alphabet or symbol is chosen.

A. Population

Genetic algorithm are [4] generally start with an initial population that is generated randomly some research has been conducted using special technique to produce a higher quality initial population. Thus an approach is designed to give the GA a good start and speed up the evolutionary process.

Here the example, it is given a sentence as an input such as “This genetic algorithm is used for searching a paragraph. It shows that the algorithm can find the matching paragraph based on input by a user.”
B. Fitness

Fitness function makes certain that evolution forward to optimization with accumulating for each individual on population.

\[ \text{Fitness} = 64(n) - E = 64(n) - \sum_{i=1}^{n} gitti \]

3. Discussion and Implementation

In this section will discuss about process and result of sentence matching using genetic algorithm. The parameters and values of genetic algorithm on this research are:

1. Population = 10, 20, 50
2. Generation = 50, 80, 100
3. Crossover probability = 0.5
4. Mutation probability = 0.05

| Table 1. 10 Population 50 Generation |
|--------------------------------------|
| Experiment | Generation | Fitness |
| 1          | 169        | 672     |
| 2          | 168        | 755     |
| 3          | 224        | 672     |
| 4          | 152        | 740     |
| 5          | 166        | 772     |
| 6          | 146        | 683     |
| 7          | 144        | 877     |
| 8          | 156        | 835     |
| 9          | 154        | 815     |
| 10         | 212        | 587     |

On Table 1 is seen that fitness value over all experiment the result is different, and this experiment is done ten times. The result on the table above is that the highest of fitness value is 877 for 144 generation, meanwhile the lowest fitness value is on generation 169 is 672.

| Table 2. 50 Population 100 Generation |
|--------------------------------------|
| Experiment | Generation | Fitness |
| 1          | 305        | 930     |
| 2          | 210        | 955     |
| 3          | 286        | 937     |
| 4          | 242        | 910     |
| 5          | 276        | 837     |
| 6          | 361        | 865     |
| 7          | 214        | 860     |
On table 3 is seen that finess value each experiment the result is different, and the experiment is done ten times. On this experiment that the highest fitness value is 955 on generation 210, whereas the lowest fitness value is on generation 276 is 837.

| Generation | Previous Work I | Previous Work II | Recent Work |
|------------|-----------------|-----------------|-------------|
|            | Pencocokan kata secara acak dengan metode algoritma genetika menggunakan program pascal | Penyusunan alfabet membentuk kata menggunakan algoritma genetika | Genetic algorithm on paragraph matching |
| 490        |                 |                 | Generation  |

On the table IV show the testing result of previous work and recent work on the same context. The different is seen on the generation when the word or the sentence is found. Here the best generation in term of founding the word or the sentence is generation 144.

4. Conclusion
Based on the discussion and implementation which have done on this research, therefore the conclusion is that it is seen that the implementation of genetic algorithm in order to match the paragraph that is inputted has done well by showing the exact paragraph as it is. The generation or iteration of searching the sentence is short than the previous researched. It is on generation 144.

Acknowledgments
Thanks to all the authors and the supervisors for all the times and supports

References
[1] Masoud, N., Jayakrishnan, R. 2017. A decomposition algorithm to solve the multi-hop Peer-to-Peer ride-matching problem. Transportation Research Part B: Methodological. Volume 99, 1-129.
[2] Nikolić M., Jović J. 2017. Implementation of generic algorithm in map-matching model. Expert Systems with Applications. Volume 72, 283-292.
[3] Saragih, R. I. E. 2017. Penyusunan Alfabet Membentuk Kata Menggunakan Algoritma Genetika, Jurnal ISD UPH. Volume 2, 63-76.
[4] Crochemore M., Iliopoulos C. S, Kundu R., Mohamed M., Vayani F. 2016. Linear algorithm for conservative degenerate pattern matching. Engineering Applications of Artificial Intelligence. Volume 51, 109-114.
[5] Abu-Aisheh Z., Raveaux R., Ramel J. Y, . 2016. Anytime graph matching. Pattern Recognition Letters. Volume 84, 215-224.
[6] Holland, J.H. 1975. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence, University of Michigan Press, Ann Arbor, Mich: USA.
[7] Razali N. M., Geraghty J. 2011. A genetic algorithm performance with different selection strategies. Proceedings of the World Congress on Engineering Vol II.
[8] Sivanandam, S.M, & Deepa, S.N. 2008. Introduction to Genetic Algorithms, Springer, Berlin: Germany.

[9] Smith, J.E. 2002. Handbook of Global Optimization. Volume 2, 275-362.

[10] Varnamkhasti, M.J, Lee, L.S, Bakar, M.R.A, & Leong, W. J. 2012. A Genetic Algorithm with Fuzzy Crossover Operator and Probability. Volume 2012, Article ID 956498, 16.

[11] Burke, E.K., Elliman D.G. Weave, R.F., “A Hybrid Genetic Algorithm for highly constrained Timetabling problems”, Proc. of 6th International Conference on the practice and Theory of Automated Timetabling, Napier University, Edinburgh, UK, 1995.

[12] K.F. Man, K.S. and Tang, S. Kwong, “Genetic Algorithms: Concept and Designs”, Springer, Chapter 1-10, Pg 1-348.

[13] Mulia, A. K., 2010. Pencocokan Kata Secara Acak Dengan Metode Algoritma Genetika Menggunakan Program Pascal, UNAND. Volume 2, 1-9.

[14] Kaur M., Kaur N. 2013. Web document clustering approaches using k-means algorithm, International Journal of Advanced Research in Computer Science and Software Engineering, Vol 3.

[15] Chandra R., Balabantaray, Sarma C., Jha M. 2013. Document clustering using k-means and k-medoids, International Journal of Knowledge based Computer Systems, Vol 1, Issue 1.

[16] Banjamin C.M Fung, Wang K., Ester M. 2003. Hierarchical document clustering using Frequent itemsets. In proceedings of SIAM International Conference on Data mining.

[17] Yunsha, Zhang G., Jiang H. 2007. Text clustering based on Lexical graph. Fourth International Conference on Fuzzy System and Knowledge Discovery.

[18] Ma H., Fan X., Jichen. 2008. An incremental Chinese text classification algorithm based on quick clustering. International Symposium on Information Processing.

[19] Guo X., Shao Z, Hua N. 2009. A hierarchical Text clustering algorithm with cognitive situation Dimensions. Second International Workshop on Knowledge discovery and Data Mining.

[20] Hu J, Xiong C., Shu J, Zhou X., Zhu J. 2009. A novel Text clustering method based on TGSOM and Fuzzy K-means, First International Workshop on Educationa Technology and Computer Science.

[21] Zang Z, Cheng H. 2008. Clustering Aggregation based on genetic algorithm for Document clustering. IEEE Congress on Evolutionary Computation.

[22] Zhu, K.Q, & Liu, Z. 2004. Population Diversity in Permutation-Based Genetic Algorithm. ECML, LNAI, pp 537-547.

[23] Alt, H., Efrat, A., Rote, G., & Wenk, C. (2003). Matching planar maps. Journal of Algorithms, 49(2), 262–283.

[24] Reeves, C., & Rowe, J. (2002). Genetic Algorithms Principles and Perspectives. Kluwer, Norwell, MA.

[25] Coley, D. A. (1999). An introduction to genetic algorithms for scientists and engineers. World scientific.