Optimization of Spaced *K*-mer Frequency Feature Extraction using Genetic Algorithms for Metagenome Fragment Classification

Arini Aha Pekuwali[1], Wisnu Ananta Kusuma[2,*] & Agus Buono[2]

[1]Department of Informatics Engineering, Faculty of Science and Engineering, Universitas Kristen Wira Wacana, Jalan R. Suprapto No. 35, Prailiu, Waingapu, Sumba Timur, 87113, Indonesia
[2]Department of Computer Science, Faculty of Mathematics and Natural Science, Bogor Agricultural University, Jalan Meranti, Kampus IPB Darmaga, Bogor 16680, Indonesia
*E-mail: ananta@apps.ipb.ac.id[2]

Abstract. *K*-mer frequencies are commonly used in extracting features from metagenome fragments. In spite of this, researchers have found that their use is still inefficient. In this research, a genetic algorithm was employed to find optimally spaced *k*-mers. These were obtained by generating the possible combinations of match positions and don't care positions (written as *). This approach was adopted from the concept of spaced seeds in PatternHunter. The use of spaced *k*-mers could reduce the size of the *k*-mer frequency feature's dimension. To measure the accuracy of the proposed method we used the naïve Bayesian classifier (NBC). The result showed that the chromosome 111111110001, representing spaced *k*-mer model [111 1111 10001], was the best chromosome, with a higher fitness (85.42) than that of the *k*-mer frequency feature. Moreover, the proposed approach also reduced the feature extraction time.

Keywords: *genetic algorithm;* k-*mers; metagenome; naïve Bayesian classifier; spaced* k-*mers.*

1 Introduction

A common approach to producing DNA sequences for studying the genetic material of organisms is to perform de novo sequence assembly from reads produced by Next Generation Sequencer (NGS) using DNA sequence assembly tools such as Velvet [1], Edena [2], and SOAP denovo [3]. These reads are obtained from a sample of the organism cultivated in the lab. Unfortunately, only about 1% of the many microorganisms in the world can be cultured [4]. The rest must be collected by taking samples directly from the environment.

Metagenomics is the study of the entire genetic information of organism samples that are directly taken from the environment, such as soil, water,

Received October 21st, 2016, 1st Revision July 9th, 2017, 2nd Revision April 27th, 2018, 3rd Revision August 23rd, 2018, Accepted for publication August 31st, 2018.
Copyright © 2018 Published by ITB Journal Publisher, ISSN: 2337-5787, DOI: 10.5614/itbj.ict.res.appl.2018.12.2.2

buildings, or waste where microbes breed [5]. Metagenomics aims to study species variations, contributes to the discovery of new genes and describes the interaction between microbes and their host [6].

Metagenomics analysis starts with deoxyribonucleic acid (DNA) sequencing on the metagenome sample. The resulting fragments contain various microorganisms because they are taken directly from the environment [5]. Such conditions may cause errors in the assembly of the metagenome fragments, called misassembly contigs. Misassembly yields interspecies chimeras [7]. To minimize the number of interspecies chimeras, binning and assembly can be performed simultaneously.

Binning is a process in which various fragments of an organism are grouped together based on their taxonomic level. There are two binning approaches, i.e. homology-based and composition-based approaches. In an homology-based approach, sequence alignment is performed between the metagenome fragments and the sequence reference that exist in the database of the National Center for Biotechnology Information (NCBI). In a composition-based approach, binning is conducted by classification or clustering using machine learning methods.

BLAST (Basic Local Alignment Search Tool) [8] and MEGAN [9] are applications that use an homology-based approach for identifying species. Meanwhile, a composition-based approach was adopted by some applications for performing metagenome fragment binning, such as PhyloPythia, which uses SVM for performing metagenome fragment classification [10], classification based on the naïve Bayesian classifier [11], and metagenome fragment clustering based on a growing self organizing map (GSOM) [12].

PhyloPythia uses *k*-mer frequency feature extraction and support vector machine (SVM). The present research used large *k* values; the minimum *k* value was 5 in view of obtaining a high accuracy percentage. Another research has been conducted using the naïve Bayesian classifier (NBC) [11]. NBC can assign next-generation sequencing reads to their taxonomic classification [13]. Feature extraction was done using *k*-mer frequencies; the *k* values ranged from 3 to 15 mers. The research concluded that the highest accuracy percentage was obtained with the use of 12 mers for 250 base pairs (bp) and 100 bp. Meanwhile, application of unsupervised learning using GSOM and *k*-mer frequency feature extraction [12] can be used to cluster short fragments of large communities.

The main problem of using the *k*-mer frequency feature is dealing with a large-dimension feature space when aiming to obtain high accuracy [14]. To solve this problem, Kusuma [15] introduced spaced *k*-mers, inspired by PatternHunter [16], to reduce the feature space dimension and improving accuracy. Based on

an exhaustive search, the optimal spaced *k*-mer feature space consisted of 192 features. Classification was conducted using SVM. Ref. [15] reports that good accuracy could be obtained, even for a small fragment length (400 bp), with an accuracy of 65.3% for genus taxon, 72% for order taxon, 78.2% for class taxon, and 82.1% for phylum taxon. For long fragments (10 Kbp), the accuracy reached more than 95% for all taxon levels.

Spaced *k*-mer feature extraction results in many model variations of match positions (1) and don't care positions (0) by using exhaustive search. Therefore, position model variations that can result in high accuracy need to be found. A genetic algorithm (GA) can be used to find the optimally spaced k-mers, which can result in higher accuracy. Hence, in this research, a GA was used to optimize the match and don't care positions in spaced *k*-mer feature extraction. GAs are widely used in solving gene selection problems [17], such as finding the most informative genes that contribute to cancer classification using computational intelligence algorithms [18].

This research aimed to find the match and don't care positions resulting in the best accuracy by using GA optimization. The second aim of this research was to know the influence of the use of don't care positions on spaced *k*-mer feature extraction.

2 Research Method

The research method consisted of 4 parts (see Figure 1):

2.1 Data Collection and Pre-processing

This research used data obtained from the NCBI database, which can be accessed via (http://www.ncbi.nlm.nih.gov/). The data format used was FASTA (*.fna). There were 19 species, which included 3 genii [19] (as shown in Tables 1 and 2). The dataset was divided into 2 parts, i.e. a training dataset, containing 10 species, and a testing dataset, containing 9 species.

Pre-processing on the training data and the testing data was performed using MetaSim [20]. MetaSim is a software application that can simulate a DNA sequencer. The sequencing simulation using MetaSim resulted in 10,000 fragments for training and 4,500 fragments for testing [15]. The length of each fragment was 500 bp. Fragments of this length have high enough accuracy to be able to classify fragments with length < 1 kbp [15].

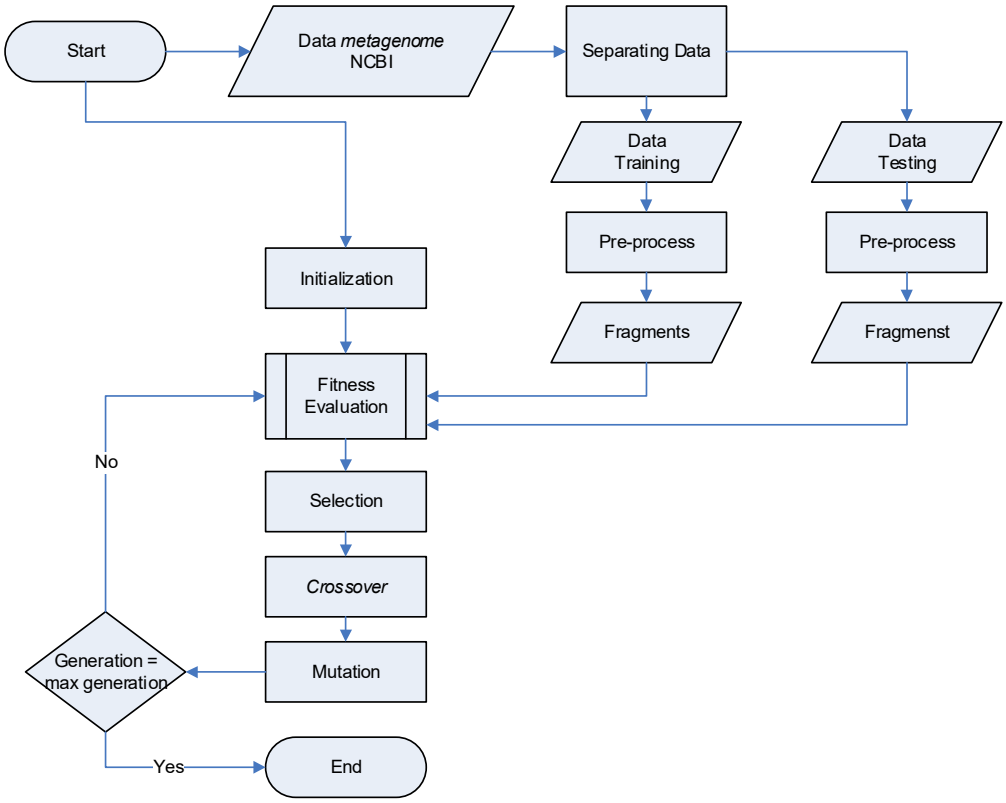

Figure 1 Research method.

Table 1 Training data.

Species	Genus	Number of Fragment	Length of Fragment
Agrobacterium radiobacter K84 ch. 2	Agrobacterium	1000	500
Agrobacterium tumafaciens str. C58 ch. Circular	Agrobacterium	1000	500
Agrobacterium vitis S4 ch. 1	Agrobacterium	1000	500
Bacillus amyloliquefaciens FZB42	Bacillus	1000	500
Bacillus anthracis str. Ames Ancestor	Bacillus	1000	500
Bacillus cereus 03BB102	Bacillus	1000	500
Bacillus pseudofarmus OF4 ch.	Bacillus	1000	500
Staphylococcus aureus subsp. Aureus JH	Staphylococcus	1000	500
Staphylococcus epidermis ATCC 12228	Staphylococcus	1000	500
Staphylococcus haemolyticus JCSC 1435	Staphylococcus	1000	500

Table 2 Test data.

Species	Genus	Number of Fragment	Length Fragment
Agrobacterium radiobacter K84 ch. 1	Agrobacterium	500	500
Agrobacterium tumafaciens str. C58 ch. Linear	Agrobacterium	500	500
Agrobacterium vitis S4 ch. 2	Agrobacterium	500	500
Bacillus thuringiensis str Al Hakam	Bacillus	500	500
Bacillus subtilis subsp. Subtilis str 168	Bacillus	500	500
Bacillus pumilus SAFR-032	Bacillus	500	500
Staphylococcus carnosus	Staphylococcus	500	500
Staphylococcus saprophyticus subsp. Saprophyticus ATCC 1530S	Staphylococcus	500	500
Staphylococcus lugdunensis HKU09-01	Staphylococcus	500	500

2.2 Optimization of Spaced K-Mer Feature Extraction Using GA

First, the GA population is initialized. The second step is feature extraction. This process results in features that are classified with the naïve Bayesian classifier (NBC). NBC is used to determine the fitness of each chromosome. After that, the chromosomes are processed to obtain the best chromosomes. Next, crossover is conducted on the selected chromosomes. Lastly, they are mutated. This process is repeated from the second to the last step, while the number of generations is smaller than or equal to the maximum number of generations.

The encoding stage produces the initial population (generation 0) of individuals on which evolution is based. Since problems differ from one another, the encoding stage is usually problem-specific [21]. Chromosome initialization using GA can be explained as follows. For instance, using $k = 12$, chromosomes are formed consisting of 12 genes (Figure 2). Using $k = 12$ this yields 4^{12} features. Therefore, the concept of spaced seeds from PatternHunter [16] was adopted to modify the *k*-mer frequency feature, getting so-called spaced *k*-mer frequencies, which consist of match positions (1) and don't care positions (0). The concept of spaced seeds has also been employed in BLASTZ [22]. Thus, using $k = 12$, the GA has a search space of 4,096 possibilities.

1	1	1	1	1	1	1	1	1	1	1	1
x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}

Figure 2 *K*-mer frequencies formed if $k = 12$.

2.3 Feature Extraction of Metagenome Fragments

In this study, feature extraction was conducted by calculating the spaced *k*-mer frequencies of metagenome fragments. This study found the optimum *k*-mer pattern that includes don't care positions. Match position (1) and don't care

position (0) models were formed from the GA initialization result. The model position is a chromosome. Don't care position (0) means allowing any base pair to fill the bit [16].

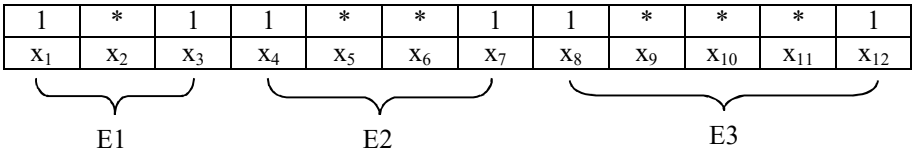

Figure 3 Ilustration of chromosome initialization using GA.

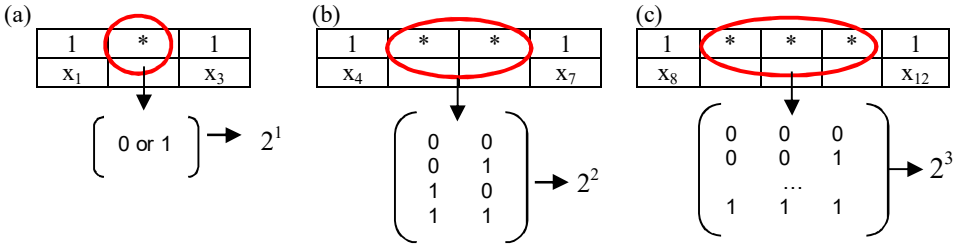

Figure 4 (a) Example of possibility formed from E1, (b) example of possibility formed from E2, (c) example of possibility formed from E3.

Figure 3 shows chromosome feature extraction, which was initialized consisting of 3 parts. The first part, E_1 (Figure 4a), consists of 3 genes with a variation possibility of 2^1. Figure 4b shows the second part, E_2, consisting of 4 genes with a variation possibility of 2^2. The third part, E_3 (Figure 4c), consists of 5 genes that have a variation possibility of 2^3. The features from the feature extraction process were formed by combining nucleotide *adenine* (A), *cytosine* (C), *guanine* (G) and *thymine* (T).

1 0 1 1 0 1 1 1 0 1 0 1

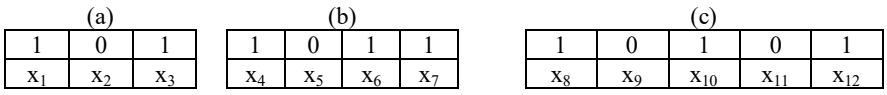

Figure 5 Ilustration of selected chromosomes.

The total number of DNA sequence combinations is calculated by using 4^k. 4 is the number of tuples (A, T, G, C), while k is the number of biner 1. For example, in (a), $k = 2$, so there are $4^2 = 16$ possible combinations. Using the

chromosome in Figure 4, the feature space dimension would be as shown in Table 3 show.

Table 3 Feature space dimension formed with chromosome 101 1011 10101.

Feature Fragment	A*A (1)	…	T*T (16)	A*AA (17)	…	T*TT (80)	A*A*A (81)	…	T*T*T (144)
F1									
F2									
…									
Fn									

2.4 Classification of Metagenome Fragments using NBC

Feature extraction and classification using NBC is important for the fitness evaluation. Chromosomes that have been formed at initialization are used to model the feature extraction process using spaced *k*-mer frequencies. Then, the resulting features are classified using NBC. The accuracy value generated by NBC is the fitness value of the chromosome used.

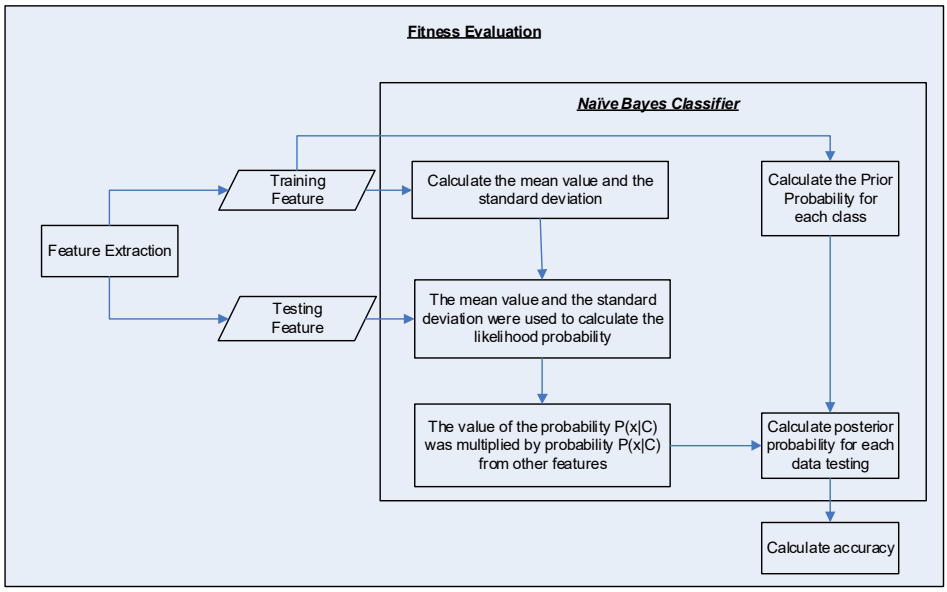

Figure 6 Fitness evaluation.

The classification method used in this research was the naïve Bayesian classifier (NBC). Bayes' theorem is the cornerstone of this method. If $x = [x_1, x_2, x_3, ..., x_n]^T$ is a feature vector consisting of a set of words with the length of the fragment; label x is one of the genomes m; C_1, C_2, C_3 are the posterior

probabilities of a particular class (C_i) associated with feature vector x, i.e. $P(C_i|x)$ [23].

$$C = \text{argmax } P(C_i|x) \tag{1}$$

According to Figure 6, the first step is to calculate the mean value and standard deviation of the data training features for each class [24]. The mean value and standard deviation were used to calculate probability $P(x_k|C_i)$.

$$P(x_k|C_i) = g\left(x_k, \mu_{C_i}, \sigma_{C_i}\right) = \frac{1}{\sqrt{2\pi}\,\sigma_{C_i}} e^{-\frac{(x_k-\mu_{C_i})2}{2\sigma_{C_i}^2}} \tag{2}$$

where $g\left(x_k, \mu_{C_i}, \sigma_{C_i}\right)$ is the Gaussian density for attribute A_k, μ_{C_i} and σ_{C_i} are the mean and standard deviation. After that, probability value $P(x_k|C_i)$ was multiplied by probability $P(x_k|C_i)$ of the other features.

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i) \tag{3}$$

Thus, probability $P(X|C_i)$ for each class is obtained. Probability $P(X|C_i)$ is multiplied by the prior probability for each class, resulting in posterior probability $P(C_i|X)$.

$$P\left(C_i \mid X\right) = \frac{P\left(X \mid C_i\right) \quad P(C_i)}{P(x)} \tag{4}$$

In order to classify an unknown sample X, $P(X|C_i)$ $P(C_i)$ is evaluated for each class C_i [24]. Sample X is then assigned to class C_i if and only if

$$\text{P(X}|C_i)\ P(C_i) > P(X|C_j)\ P(C_j) \textit{ for } 1 \le j \le m, j \ne i \tag{5}$$

The accuracy of the classification result can be found by using following formula:

$$\text{Accuracy} = \frac{\sum true\_data\_testing}{\sum data\_testing} \text{ x } 100\% \tag{6}$$

The obtained accuracy value is the fitness value of the chromosome of the GA initialization result.

2.5 Genetic Algorithm

Genetic algorithms are widely used to solve hard optimization problems [25]. GAs have high solving speed. GA operators are for example genes, chromosomes and populations [25]. In GAs, the population of a candidate

solution to an optimization evolves toward better solutions [26]. A genetic algorithm was used in this research to optimize chromosomes containing match (1) and don't care (0) positions. The population size was set to 20 chromosomes. The genetic operators applied in a simple GA to test the performance of our approach are described in Table 4 [27,28].

Table 4 Genetic parameters for simple GA .

Selection operator	Roulette Wheel
Crossover operator	One cut point
Mutation operator	One mutation at a random position
Crossover probability	0.65; 0.70; 0.75
Mutation probability	0.050; 0.075; 0.100
Maximum generation	50
Elitism	1

3 Results and Analysis

The GA optimized the chromosome that was used as a pattern in feature extraction. The form of the chromosome used in this research was matched with the one in Figure 3. There were 64 chromosome combinations that were formed and a fitness check was conducted on each of them. The search technique must find a good trade-off between exploration and exploitation within the selection mechanism in order to find the global optimum [29]. Exploration means that poor solutions must have a chance to go to the next generation, while exploitation means that good solutions go to the next generation more frequently than poor solutions. We conducted the experiment 9 times. Figure 7 shows that Experiment 3, which used $Pc = 0.65$ and $Pm = 0.1$, performed the best because it managed to find the global optimum point at 62.5% of the search space.

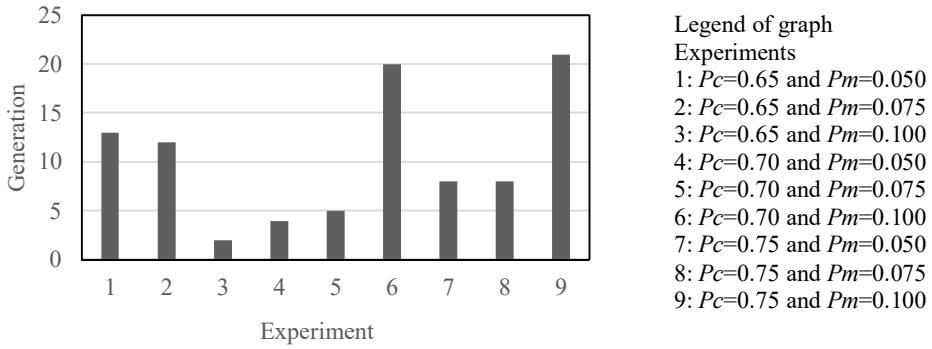

Figure 7 GA experiment.

Figure 8 shows the highest fitness graph for each generation. It can be seen that the GA with crossover probability 0.65 and mutation probability 0.1 managed to find the global optimum point in the second generation. Chromosome 111111110001, which formed the pattern [111 1111 10001], was selected as the best chromosome, with fitness 85.42.

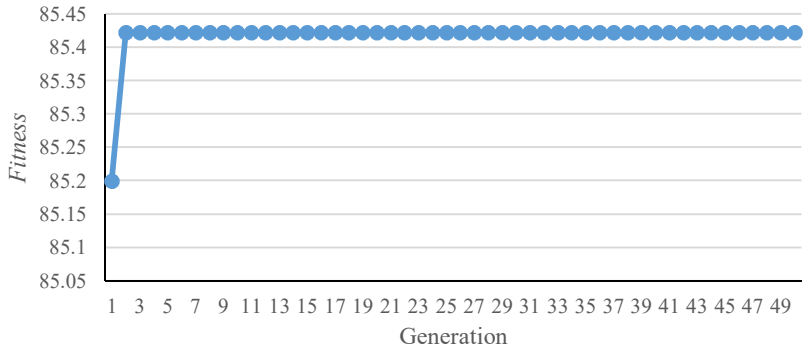

Figure 8 Highest fitness for each generation.

3.1 Confusion Matrix

Table 5 shows that the amount of test data used was 5000 fragments. This is known by summing the numbers listed in the matrix. The first line shows that from the 1500 fragments of the Agrobacterium genus, 1462 fragments were correctly classified as Agrobacterium genus and 38 fragments were incorrectly classified as Bacillus genus. The second line shows that 1116 fragments were correctly classified as Bacillus genus, 40 fragments were incorrectly classified as Agrobacterium genus and 344 fragments were incorrectly classified as Staphylococcus genus.

Table 5 Confusion matrix of chromosome 111111110001.

Prediction **Actual**	***Agrobacterium***	***Bacillus***	***Staphylococcus***
Agrobacterium	1462	38	0
Bacillus	40	1116	344
Staphylococcus	0	234	1266

The third line shows that 1266 fragments were correctly classified as Staphylococcus genus and 234 fragments were incorrectly classified as Bacillus genus. The bacillus genus fragments incorrectly classified as staphylococcus genus and the staphylococcus genus fragments incorrectly classified as bacillus genus were incorrectly classified because the bacillus and staphylococcus genii are both derived from the same Bacillales order.

3.2 Comparison of *K*-mer Frequency with Spaced K-Mers

Chromosome 111111110001, forming the pattern [111 1111 10001] and producing 336 features [AAA … TTT AAAA … TTTT A***A … T***T], was compared to chromosome 111111111111, forming pattern [111 1111 11111] and producing 1344 features [AAA … TTT AAAA … TTTT AAAAA … TTTTT]. Chromosome 111111110001 represents the spaced *k*-mer pattern and chromosome 111111111111 represents the *k*-mer pattern.

Chromosome 111111110001 had the best fitness (85.42), which was higher than that of chromosome 111111111111 (85.15). Thus, the spaced *k*-mer pattern yielded by GA improved the classification accuracy (shown by the fitness values).

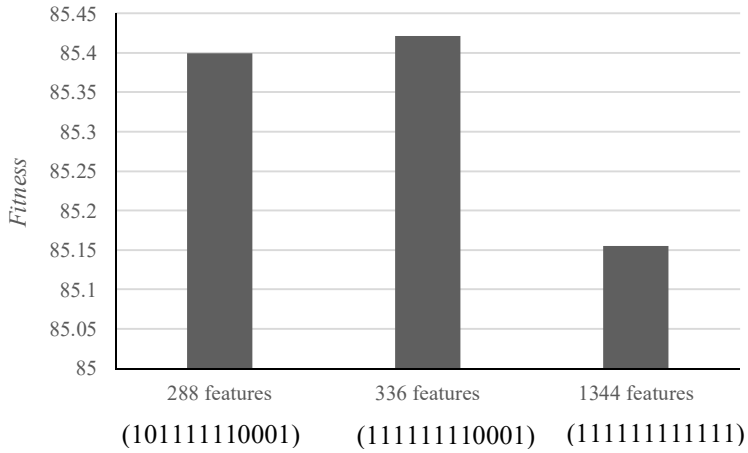

Figure 9 Comparison of fitness values.

Moreover, chromosome 101111110001, yielding the pattern [101 1111 10001] and producing 228 features [A*A … T*T AAAA … TTTT A***A … T***T], was compared to chromosome 111111111111 producing 1344 features. Figure 10 shows that the execution time of processing chromosome 101111110001 was around 56 minutes. This was faster than for chromosome 111111111111, which took an execution time of 68 minutes for finishing the task. The spaced *k*-mers also reduced the number of feature dimensions, hence accelerating the execution time.

The drawback of the proposed method is that the experiment for performing 50 generations took 10 days. Hence, the application needs to be developed further to work in parallel so the execution time can be shortened. The overall process of the application, from initialization to fitness evaluation, took a large amount of time, as shown in Figure 10.

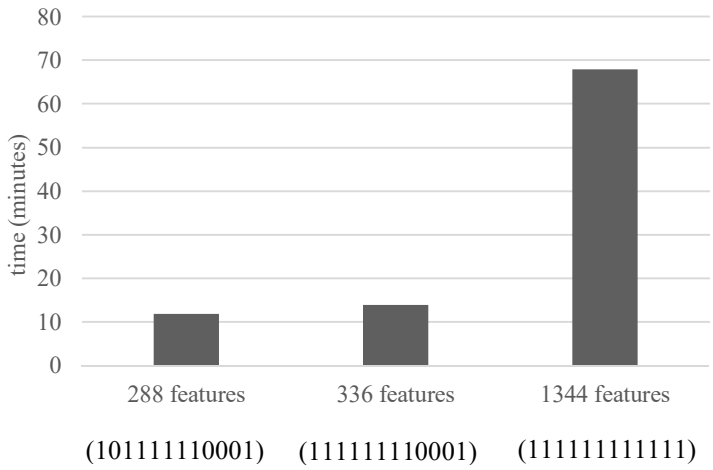

Figure 10 Comparison of execution time.

This method uses a chromosome selector, with the aim to avoid repetition of feature extraction that has already been done in the previous generation. However, the existing selector procedure can only compare the *n*-th generation with the *n1*-th generation. Therefore, a chromosome procedure is required that can compare the chromosome's *n*-th generation with other previous generations. Thus, feature extraction and classification do not need to be done repeatedly. The genetic algorithm has a high solving speed in the early solving period [30].

4 Conclusion and Future Work

Based on this study it can be concluded that the genetic algorithm managed to find the global optimum point with a fitness of 85.42. The best chromosome was 111111110001, producing 336 features. Using spaced *k*-mers improved the accuracy of the classification and also reduced the execution time.

Future work can be conducted by performing parallel programming for screening to generate chromosomes by comparing chromosome *n* of generation *m* with whole chromosomes that have been raised in previous generations. This is expected to reduce the execution time during the training phase using GA.

To further validate the efficiency of the proposed method in the classification of short metagenomic fragments, we plan to use a real dataset, such as the Sargasso Sea dataset or metagenomic data from an acid mine.

References

[1] Zerbino, D.R. & Birney, E., *Velvet: Algorithms for De Novo Short Read Assembly Using de Bruijn Graph.* Genome Research, **18**(5), pp. 821-829, 2008.

[2] Hernandez, D., François, P., Farinelli, L., Osterås, M. & Schrenzel, J., *De novo Bacterial Genom Sequencing: Millions of Very Short Reads Assembled on a Desktop Computer,.* Genome Research, **18**(5), pp. 802-809, 2008.

[3] Li, R., Zhu, H., Ruan, J., Qian, W., Fang, X., Shi, Z., Li, Y., Li, S., Shan, G., Kristiansen, K., Li, S., Yang, H. & Wang, J., *De novo Assembly of Human Genomes with Massively Parallel Short Read Sequencing,* Genome Research, **20**(2) pp. 265-272, 2009.

[4] Wu, H., *PCA-based Linear Combinations of Oligonucleotide Frequencies for Metagenomic DNA Fragment Binning Proc.*, IEEE Symp. Computational Intelligence in Bioinformatics and Computational Biology (CIBCB '08), IEEE Press, pp. 46-53, doi:10.1109/CIBCB.2008.4675758, 2008.

[5] Bouchot, J.L., Trimble, W.L., Ditzler, G., Lan, Y., Essinger, S. & Rosen, G., *Advances in Machine Learning for Processing and Comparison of Metagenomic Data,* Available on: http://www.math.drexel.edu/~jb3455/publications/preprints/metagenomics_BC.pdf, 2013. [download September 15, 2014]

[6] Thomas, T., Gilbert, J. & Meyer, F., *Metagenomics – A Guide from Sampling to Data Analysis*, Microbial Informatics and Experimentation, 2012.

[7] Meyerdierks, A. & Glockner, F.O., *Metagenome Analysis*, Advances in Marine Genomics, 2010.

[8] Altschul, S., Gish, W., Miller, W., Myers, E., & Lipman, D., *Basic Local Alignment Search Tool,* Journal of Molecular Biology, **215**(3) pp. 403-410, 1990.

[9] Huson, D.H., Auch, A.F. & Schuster, S.C., *MEGAN Analysis of Metagenomic Data*, Genome Research, **17**(3) pp. 337-386, 2007.

[10] McHardy, A.C., Martin, H.G., Tsirigos, A., Hugenholtz, P. & Rigoutsos, I., *Accurate Phylogenetic Classification of Variable-Length DNA Fragments*, Nature Methods, **4**(1) pp. 63-72, 2007.

[11] Rosen, G., Garbarine, E., Caseiro, D., Polikar, R. & Sokhansanj, *Metagenome Fragment Classification Using N-mer Frequency Profiles*, Advances in Bioinformatics, 2008.

[12] Overbeek, M.V., Kusuma, W.A. & Buono, A., *Clustering Metagenome Fragments Using Growing Selforganizing Map*,Advanced Computer

Science and Information Systems (ICACSIS), 285-289, DOI: 10.1109/ICACSIS.2013. 6761590, 2013.

[13] Rosen, G.L., Reichenberger, E.R. & Rosenfeld, A.M., *NBC: The Naive Bayes Classification Tool Webserver for Taxonomic Classification of Metagenomic Reads*, Bioinformatics, **27**(1), pp. 127-129, 2011.

[14] Gsponer, S., Smyth, B. & Ifrim, G., *Efficient Sequence Regression by Learning Linear Models in All-Subsequence Space*, Insight Centre for Data Analytics, 2017.

[15] Kusuma, W.A., *Combined Approaches for Improving the Performance of Denovo DNA Sequence Assembly and Metagenomic Classification of Short Fragments from Next Generation Sequencer* [Dissertation], Tokyo (JP): Tokyo Institute of Technology, 2012.

[16] Ma, B., Tromp, J. & Li, M., *PatternHunter: Faster and More Sensitive Homology Search*, Bioinformatics, **18**(3) pp. 440-445, 2002.

[17] Alomari, O.A., Khader, A.T., Al-Betar, M.A. & Abualigah, L.M., *Gene Selection for Cancer Classification by Combining Minimum Redundancy Maximum Relevancy and Bat-inspired Algorithm*, Journal of Data Mining and Bioinformatics, **19**(1) pp. 32-51, 2017.

[18] Alomari, O.A., Khader, A.T., Al Betar, M.A. & Abualigah, L.M., *MRMR BA: A Hybrid Gene Selection Algorithm for Cancer Classification*, Journal of Theoretical and Applied Information Technology, **95**(12), pp. 2610-2618, 2017.

[19] Kusuma, W.A. & Akiyama, Y., *Metagenome Fragments Classification Based on Characterization Vectors,* Proceedings of International Conference on Bioinformatics and Biomedical Technology, Sanya China, pp. 50-54, March, 2011.

[20] Richter, D.C., Ott, F., Auch, A.F., Schmid, R. & Huson, D.H., *MetaSim: a Sequencing Simulator for Genomics and Metagenomics*, PloS One, **3**(10) pp. 1-12, 2008.

[21] Goh, K. S., Lim, A. & Rodrigues, B., *Sexual Selection for Genetic Algorithms*, Artificial Intelligence Review, **19,** pp. 123-152, 2003.

[22] Schwartz, S., Kent W.J., Smit, A., Zhang, Z., Baertsch, R., Hardison, R.C., Haussler, D. & Miller,W., *Human-Mouse Alignment with BLASTZ*, Genome Research, **13**, pp. 103-107, 2003.

[23] Mitchell, T.M., *Machine Learning,* US: Mc Graw-Hill Science/ Engineering/ Math, 1997.

[24] Han, J. & Kamber, M., *Data Mining: Concepts and Techniques*, San Francisco (US): Morgan Kaufmann Publishers, 2001.

[25] Abualigah, L.M., Khader, A.T. & Al-Betar, M.A., *Unsupervised Feature Selection Technique Based on Genetic Algorithm for Improving the Text Clustering*, 7th CSIT (International Conference on Computer Science and Information Technology), pp. 1-6, 2016.

[26] Abualigah, L.M. & Hanandeh, E.S., *Applying Genetic Algorithms to Information Retrieval using Vector Space Model,* IJCSEA, **5**(1), pp. 19-28, 2015.

[27] Angelova, M. & Pencheva, T., *Tuning Genetic Algorithm Parameter to Improve Converge Time*, International Journal of Chemical Engineering. doi: 10.1155/2011/646917, 2011.

[28] Gen, M. & Cheng, R., *Genetic Algorithms and Engineering Design*, New York: John Wiley & Sons, Inc., 1997.

[29] Razali, N. M. & Geraghty, J., *Genetic Algorithm Performance with different Selection Strategies in Solving TSP,* Proceedings of the World Congress on Engineering, London (UK), **II**, July 6-8, 2011.

[30] Zhang, X. & Liu, S., *Image Edge Feature Extraction and Refining Based on Genetic-Ant Colony Algorithm*, Telkomnika, **3**(1), pp. 118-127, 2015.