Seminar Nasional Aplikasi Teknologi Informasi 2005 (SNATI 2005) *ISBN: 979-756-061-6*
Yogyakarta, 18 Juni 2005

CLASSIFICATION OF INDONESIAN SPEECH INTO VOICED-UNVOICED-SILENCE USING EVOLVING FEEDFORWARD NEURAL NETWORKS

Suyanto
Jurusan Teknik Informatika Sekolah Tinggi Teknologi Telkom
Jl. Telekomunikasi, Dayeuh Kolot, Bandung 40257
E-mail: suy@stttelkom.ac.id

Abstract

This paper describes a system to classify Indonesian speech into voiced-unvoiced-silence (VUS). In this system, a speech of 16 KHz is segmented into frames of 10 milliseconds with overlap of 20%. Next, each frame is characterized using 3 features in time domain: frame energy (E), level crossing rate (LCR) and differential level crossing rate (DLCR). Furthermore, each frame is classified using an Evolving Feedforward Neural Network (EFNNs), which is Feedforward Neural Network (FNNs) that be trained using evolutionary algorithms (EAs). Finally, the classified frames are concatenated to get a right VUS classification. The training data is combination of 18 consonants and 7 vowels from a single speaker. Whereas validation set and testing data is developed from 25 word speeches represent all the combination of consonants and vowels. Computer simulation shows that the best FNNs architecture is 3-10-3 (3 inputs, 10 hidden unit, and 3 output units) and the appropriate number of training data is 150. It gives a total accuracy of 0.7366, where the accuracies for voiced, unvoiced, and silence respectively are 0.6206, 0.6428, and 0.9626. Since the accuracies for voiced and unvoiced are very low, then the whole VUS system is poor, even a filtering procedure has been applied.

***Keywords**: indonesian speech, voiced-unvoiced-silence classification, evolving feedforward neural network*

1. Introduction

Voiced-unvoiced-silence classification is one of important problems in speech processing area. In phonetically speech recognition, information about voiced, unvoiced, and silence is the main problem. Indonesian is a language with very simple rules of phonetics. There are only 18 consonants (unvoiced) and 7 vowels (voiced) [8]. It is much simpler than English. More than 230 millions Indonesian people use this language. Thus, it is very important to develop a speech recognition system for this language. Many applications can be created using this speech recognition system.

This research focuses on classification of Indonesian speech into voiced, unvoiced, and silence. Research by Mark Greenwood and Andrew Kinghorn showed that using two features, Signal Energy Rate (SER) and Zero-Crossing Rate (ZCR), yields average accuracy 65% for 10 English speeches [1]. It is caused by overlapping of the two time domain features. Using an additional feature in frequency domain, wavelet packet of Daubechies 8, improved the accuracy of VUS classification to 90.2% for four Indonesian word speeches [3]. Wavelet packet showed good performance of feature extraction, but it is very time consuming. In this research, 3 features in time domain, frame energy (E), level crossing rate (LCR) and differential level crossing rate (DLCR), are used for time reason.

Research objectives:

1. Study three time domain features of VUS;
2. Develop a VUS system using the features;
3. Investigate performance of the system.

2. VUS system

VUS system consists of four stages: normalization, segmentation, feature extraction, and classification, as illustrated by figure 2.1 below.

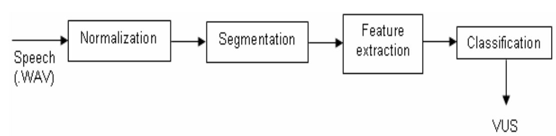


Figure 2.1 Block diagram of the VUS System.

Speech is recorded (the format is .wav) using frequency sampling 16 KHz and 16 bit level of quantization. To eliminate the amplitude difference in recording phase, the speech is normalized into range [-1, +1]. Furthermore, speech is segmented into frames of 10 milliseconds (160 samples) with overlap 20% to increase the accuracy. Each frame is extracted to be three features, E, LCR, and DLCR. In the last stage, each frame is classified using EFNNs.

2.1 Feature Extraction

The three features used in this research are described below:

1. Frame Energy (E)

$$ENERGY(k) = 10 * \log_{10}\left(\sum_{n=0}^{N-1} x(n)^2 + 1\right),$$

where k represents the analysis frame and N is the length of the analysis frame and $X(n)$ is input speech signals without preemphasis.

2. Level Crossing Rate (LCR)

$$LCR(k) = \sum_{n=0}^{N-1} \mathrm{sgn}(n)$$

$$\mathrm{sgn}(n) = \begin{cases} 1 & if\,[(x(n) - lcr\_level) * (x(n+1) - lcr\_level) < 0] \\ 0 & otherwise \end{cases}$$

where *lcr_level* represents the level defined in the level crossing rate and sgn(n) becomes 1 if the speech signal crosses the predefined level. In our case, this value is set as the median value of the samples from the 100 ms silence region. From an observation over the training data, I get the *lcr_level* of 0.0297.

3. Differential Level Crossing Rate (DLCR)

$$LCR(k) = \sum_{n=0}^{N-1} \mathrm{sgn}(n)$$

$$\mathrm{sgn}(n) = \begin{cases} 1 & if\,[(dx(n) - dlcr\_level) * (dx(n+1) - dlcr\_level) < 0] \\ 0 & otherwise \end{cases}$$

$$dx(n) = x(n) - x(n-1)\,,$$

where *dlcr-level* of 0.0297 is obtained by the same method used in 2 above.

2.2 Classification

In this stage, I use EFFNs that is an FFNs that be trained using Evolution Algorithms (EAs). FNNs is very popular neural network. It stores knowledge and experience of learning efficiently into a number of neurons. FNNs is time consuming in training process since it needs many iterations. But, after training process the trained FNNs gives a high speed computation since it needs only one calculation (no iteration).

The EAs 'randomly' manipulates binary data based on evolution and biology theory. This algorithm is suitable for very complex problems with 'infinite' solution space. The EAs can be used to train FNNs simply by representing weights and biases into a chromosome. In this problem, I use a chromosome that contains binary numbers. In this case, I use 30 bits for each weight or bias. Thus, for FNNs with structure 3-10-3 (3 inputs, 10 hidden units, and 3 output units), I have a chromosome that contains (40+33) x 30 = 2190 bits (genes). Next, each chromosome will be decoded into an individual that contains real numbers (each real number represents a weight or a bias). To measure the quality of individual, I use a fitness function based on mean absolute error (MAE) over a given training data. The fitness function is

$$f = \frac{1}{MAE}.$$

Furthermore, a population that contains a particular number of individuals will evolve based on evolution theory (selection and replacement) and biology theory (crossover and mutation). For simplicity, I use standard EAs with roulette wheel selection using linear fitness ranking, one point crossover, and elitism (to keep the best individual).

3. Training Data, Validation Set, and Testing Data

In Indonesian, there are 18 consonants: b, c, d, f, g, h, j, k, l, m, n, p, r, s, t, w, y, and z, and 7 vowels: a, e, ê, i, o, õ, and u [8]. Thus, I developed the training data using combinations of the consonants and vowels as described by Table A.1 in appendix A. Segmentation process, manually, for all the speeches yields 45,727 frames that divided into three classes: 23,858 voiced frames, 3,353 unvoiced frames, and 18,516 silence frames. Each frame consists of 160 samples. Extraction process for a frame yields a pattern that consists of 3 features (as described in section 2.1). Thus, I have 45,727 patterns as training data (figure 3.1).

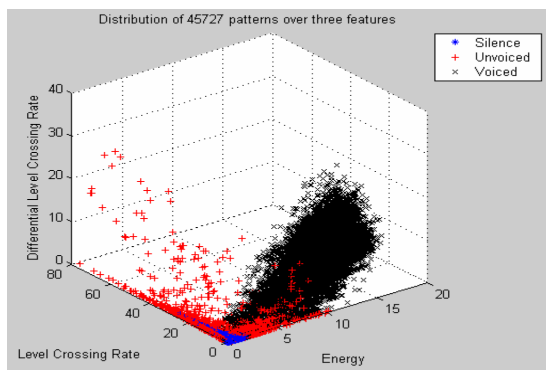


Figure 3.1 The distribution of training data over the three features.

Since the number of training data is too large, then I developed some small training data by selecting patterns from each class. It is very difficult to develop a good training data that represent all the patterns. Hence, I simply use a random procedure. Since EAs and BP are very time consuming, then I decide to develop 3 groups of training data:

- 150 patterns (50 for each class)
- 300 patterns (100 for each class)
- 1500 patterns (500 for each class)

By using the same way as in training data, I developed testing data using speech data described by Table A.2 in appendix A. After segmentation and extraction, I get 32,116 patterns that are divided into three classes: 19,144 voiced, 2,223 unvoiced, and 10,749 silences. I developed a validation set by randomly selecting 1000 patterns for each class, so that I get 3000 patterns. This validation set is used to

ISBN: 979-756-061-6

see the capability of neural network in generalization of unseen data. Next, I decided that all the 32,116 patterns are used as testing data to measure the performance of trained neural network. The testing data is illustrated by figure 3.2 below.

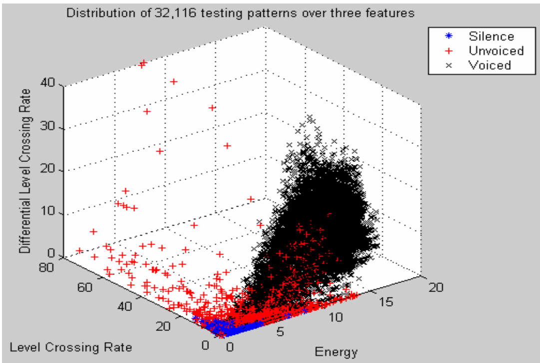


Figure 3.2 The distribution of testing data over the three features.

4. Simulation Results

In this simulation, I use EAs with roulette wheel selection using linear fitness ranking and one point crossover. To simplify the problem, I use fixed parameter values: population size of 100 individuals, each weight and bias are represented by 30 bits, crossover probability of 0.8, and mutation probability of 0.001. These values are found by trial and errors in a few experiments. This algorithm is time consuming. Using 100 individuals means I have to calculate 100 error calculations (one calculation per individual) in each generation.

4.1 Training results

Firstly, I do an experiment to find an appropriate FNNs structure and the number of training data. I do this using only 1000 generations to safe time. The figures 4.1 and 4.2 below show that FNNs with 10 hidden units give better results than FNNs with of 20 hidden units. The figures also show that using 150 training data gives the lowest mean absolute error (MAE).

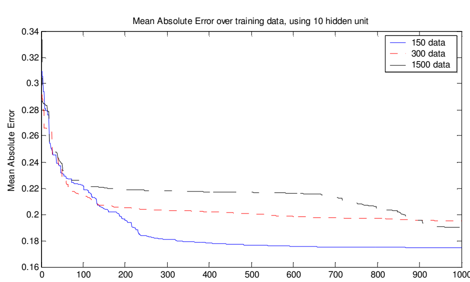


Figure 4.1 MAE of 10 hidden units FNNs over various numbers of training data.

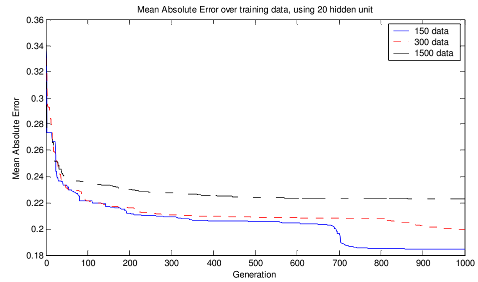
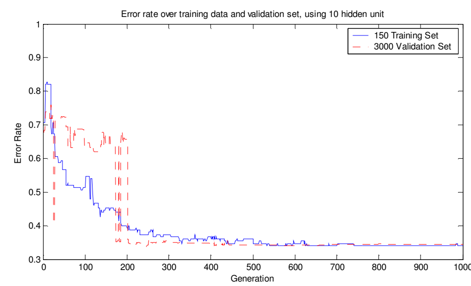
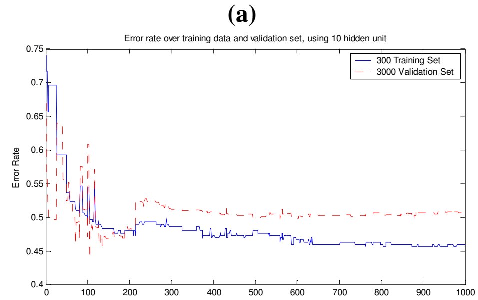


Figure 4.2 MAE of 20 hidden units FNNs over various numbers of training data.

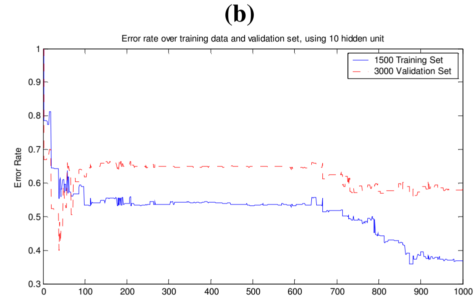
Next, to make sure that the appropriate number of training data is 150 data, I check its capability of generalization using 3000 data in validation set. Figure 4.3 shows that using 150 training data give the lowest error rate for validation set.



(a)



(b)



(c)

Figure 4.3 Error rate over training data and validation set. Using 150 training data gives the lowest error rate of around 0.35 (a). Using 300 training data gives error rate of around 0.45 (b). And using 1500 training data gives error rate of around 0.38 (c).

 ISBN: 979-756-061-6

Based on the results above, I train the FFNs use 150 training data for more generations (10,000). The results are shown by the two figures 4.4 and 4.5 below. In figure 4.5, the error rate fluctuates over validation set. This happen when error rate over training data reduce sharply in generation close to 4000. But, finally, the FFNs can generalize the 3000 validation set with error rate around 0.24. This phenomenon is a characteristic of EAs that sometimes find much better individual (after crossover). This individual, of course, much better for training set (error rate reduce sharply in generation close to 4000), but it could be very bad (too over fit) for validation set (error rate increase sharply greater than 0.5 in generation close to 4000).

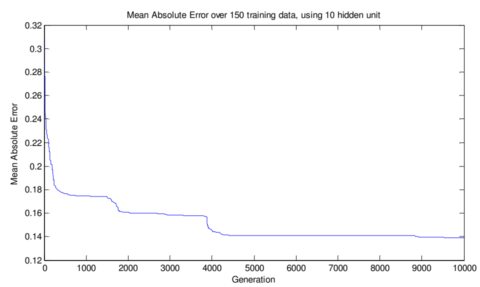


Figure 4.4 MAE for 10 hidden units FFNs using 150 training data.

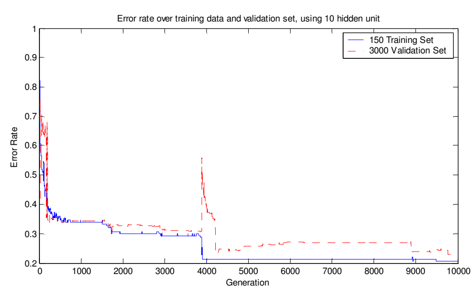


Figure 4.5 Error rate over 150 training data and 3000 validation set.

4.2 Testing results

Using results from figure 4.4, I test the trained FFNs to 32,116 testing data (i.e. 19,144 voiced, 2,223 unvoiced, and 10,749 silences). The complete results are as follow:

- Accuracy for voiced is 0.6206
- Accuracy for unvoiced is 0.6428
- Accuracy for silence is 0.9626
- Total accuracy is **0.7366**

4.3 VUS System Testing

Testing results of the VUS system shows that there are some one-frame segments are misclassified (figure 4.6). Hence, a filtering procedure is simply applied to eliminate the one-frame segments. Using this procedure I get a better result is illustrated by figure 4.7. The procedure changes sequences:

SSSUSSS → SSSSSSS,
SSSVSSS → SSSSSSS,
UUUVUUU → UUUUUUU, etc.

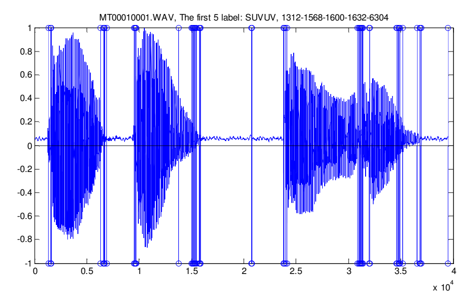


Figure 4.6 Result of VUS system without filtering. There is very narrow voiced segment (one frame) lies between unvoiced.

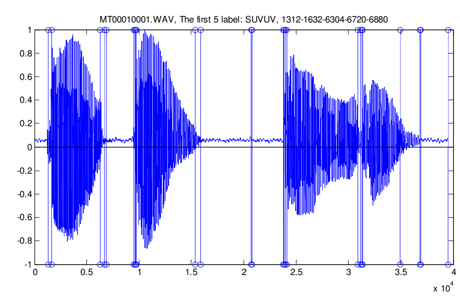


Figure 4.7 Result of VUS system after filtering. There are some one-frames eliminated.

5. Conclusions

Training and testing data show that there are many overlaps among the three time domain features. The training of EFNNs shows that it is difficult to define the number of training data needed to make the trained EFNNs has capability of generalization over the testing data. Depend only on random procedure, the optimal number of training data is 150.

From the experiment, the best architecture of EFNNs is 3-10-3 (3 inputs, 10 hidden units, and 3 output units). The VUS system gives a total accuracy of 0.7366, where accuracies for voiced, unvoiced, and silence respectively are 0.6206, 0.6428, and 0.9626. Since the accuracies for voiced and unvoiced are very low, then the whole VUS system is poor, even a filtering procedure has been applied.

6. Future Work

Using three time domain features, we can safe much time. Unfortunately, the total accuracy is very low, only 0.7366. The problem could be the three features are still not enough to distinguish three classes of speech. There are many overlaps in the features. The other possible problem could be the training data is not rich enough to generalize the testing data.

To solve the problem, we can find other alternative time domain features. We can also try to use frequency domain features, but we need a smart procedure to reduce time processing. The other work we can do is finding a particular procedure to select representative training data (not only random).

Bibliography

[1] Greenwood, Mark et al. 2001, SUVing: *Automatic Silence /Unvoiced/Voiced Classification of Speech*, United Kingdom: University of Sheffield.

[2] Youngjoo Sub et al, 1998. *Improving Speech Recognizer by Broader Acoustic- Phonetic Group Classification*. Technical report, ETRI, 161 Kajong-Dong, Yusong-Gu, Taejon, Korea.

[3] Suyanto, 2002. *Indonesian Voiced-Unvoiced-Silence Classification*. TELEKOMUNIKASI Journal, April 2002, Vol. 7 No. 1.

[4] Suyanto et al, 1999, *Speech Recognition of Indonesian Syllable with Phonemes as Its Basic Components*, Indonesia: Proceeding Industrial Electronic Seminar'99 Surabaya.

[5] Xiong, Zixiang et al. 1996, *Flexible Tree-structured Signal Expansions Using Time-varying Wavelet Packets*, IEEE Transaction on *Signal* Processing.

[6] Èmejla, Roman et al. 1999, *Adaptive Filtering for Vowel Description*, Prague: Czech Technical University, Department of Circuit Theory.

[7] Ahmed M. Abdelatty Ali. 2000, *Acoustic-Phonetic Feature-Based Signal Processing for Automatic Speech Recognition*: Brief Results, United State of America: Dept. of Electrical Engineering, University of Pennsylvania.

[8] Hasan Alwi et al. 1998, *Tata Bahasa Baku Bahasa Indonesia* (*Indonesian Grammar*). Indonesia: Balai Pustaka Jakarta.

[9] Mitchel, M et al. 1996. '*An introduction to genetic algorithms*'. MIT Press.

[10] Cain, Bibb et al. 1990. An improved Probabilistic Neural Network and Its Performance Relative to Other Models. SPIE, Vol. 1294 Application of Artificial Neural Networks, 1990.

Appendix A
Training and Testing Data

Table A.1 Speech data as training data

No	*Speech*	*Manually defined Indices of VUS segments*
1.	ba – be – bê – bi – bo – bõ – bu	2381 2553 9326 13710 13914 19958 24911 25102 31577 35408 35613 41732 45028 45262 51579 55212 55401 62534 66100 66294 71520
2.	ca – ce – cê – ci – co – cõ – cu	2727 3049 9809 14221 14664 21705 25592 26000 32894 37050 37508 43687 47566 47953 54905 59247 59566 67211 71628 71960 77892
3.	da – de – dê – di – do – dõ – du	1282 1450 8393 12040 12201 19044 23238 23405 30115 34414 34598 40700 44754 44916 51435 55609 55774 62866 67300 67469 73097
4.	fa – fe – fê – fi – fo – fõ – fu	2948 6213 12342 16808 20096 26833 31287 34881 41031 45893 49205 55025 59063 62202 68084 71410 75228 81100 85013 87876 92533
5.	ga – ge – gê – gi – go – gõ – gu	1376 1725 8104 11988 12389 18757 22850 23410 29543 32680 33209 39000 43644 44195 50098 53342 53797 60238 62824 63325 68335
6.	ha – he – hê – hi – ho – hõ – hu	3206 4960 10271 15057 16535 22352 28614 30271 35835 42451 44095 49702 54663 56226 61459 69075 69962 75488 79215 80613 86879
7.	ja – je – jê – ji – jo – jõ – ju	1329 1739 8509 10628 11186 17228 20956 21549 27455 31508 32402 37987 41425 41833 47403 51045 51711 57622 60694 61276 66756
8.	ka – ke – kê – ki – ko – kõ – ku	1764 1952 8825 14478 14786 21489 27616 27831 34710 39884 40280 46174 51722 51944 58706 64207 64407 72032 77596 77857 82654
9.	la – le – lê – li – lo – lõ – lu	1760 2353 9264 13899 14264 21894 26271 26673 33911 38062 38438 45732 50615 50931 57795 64090 64401 71504 76456 76801 83049
10.	ma – me – mê – mi – mo – mõ – mu	2108 3243 9597 16181 17177 22928 29687 31099 36409 42176 43640 49095 54558 55635 61083 66754 67930 74669 79920 80963 86575
11.	na – ne – nê – ni – no – nõ – nu	1923 3082 9427 14121 15270 21444 26929 27960 33997 38235 39746 44951 50493 51446 57475 61642 62750 69783 74089 75328 81537
12.	pa – pe – pê – pi – po – põ – pu	1929 2090 8306 12610 12803 18279 22987 23368 29027 33501 33698 39379 42842 43268 48872 53867 54148 60011 64593 64752 70478
13.	ra – re – rê – ri – ro – rõ – ru	1198 2028 7796 11666 12431 18722 22459 23597 29943 34659 35356 41789 46519 47340 54168 59232 59954 65512 68898 69804 75984
14.	sa – se – sê – si – so – sõ – su	2524 6234 12390 15973 19883 25639 29994 33114 37880 44095 46882 52259 56133 60060 64934 70090 73175 79588 82235 84788 90073
15.	ta – te – tê – ti – to – tõ – tu	1850 2033 8981 14592 14797 21980 29303 29499 35995 41065 41263 46510 51918 52128 57960 65295 65501 72040 79158 79371 84879
16.	wa – we – wê – wi – wo – wõ – wu	2138 2868 8969 14695 15372 22488 28128 28708 34984 39243 39964 45794 51143 52373 58924 64047 65030 72112 78159 78621 85098
17.	ya – ye – yê – yi – yo – yõ – yu	3712 10995 16505 16923 24234 29955 30697 38085 44864 46204 53890 59378 60623 67694 73914 74573 83423 88581 89093 96181
18.	za – ze – zê – zi – zo – zõ – zu	2975 5146 9298 14973 17320 23807 30361 32601 36925 42769 44877 51038 57363 59041 64827 71120 73002 77758 84454 87169 93336

Table A.2 Speech data for Validation Set and Testing Data

No	*Speech*	*Manually defined Indices of VUS segments*
1.	Kota baru	1267 1602 6509 9320 9684 15895 23769 24037 30877 32030 36768
2.	Sore hari	2074 5193 10096 10963 16978 24604 25739 30549 31337 36382
3.	Kêreta kêmana	1630 1844 8138 11582 12272 18828 22689 22920 29604 32792 33103 39741 41000 41719 48344 50996 51603 56075
4.	Ini tali	1644 9200 10240 12297 18431 23307 23500 30305 32795 34457 39541
5.	Biro toko	1957 2230 7808 10947 11798 18057 22171 22367 27821 30203 30480 35398
6.	Rõti rõna	2799 3939 10775 14544 14842 20847 25981 26678 32470 36301 37087 42228
7.	Batu kuda	1112 1299 7636 10631 10845 17450 21974 22285 28161 31341 31530 37620
8.	Baca bêrita	1302 1547 7270 10195 10639 16797 21358 21603 27992 32030 33403 39241 43364 43605 47957
9.	Cari acara	2257 2673 8515 11245 12088 18847 22934 29312 32728 33212 39599 42542 43826 49912
10.	Dari abadi	1756 1931 7597 10878 12085 17890 23194 29682 33500 33696 40146 43287 43462 48626
11.	Fita fana	3918 6608 12594 16634 16849 23137 26710 29914 36302 39288 40506 45564
12.	Ragi sagu	2667 2977 9822 13803 14482 20996 26314 28209 34638 36725 37413 43322
13.	Hara tahu	2173 3258 9692 11971 13214 19019 23627 23827 29715 33354 34068 39832
14.	Jari jika	1521 2034 7161 8790 9783 16343 20351 21136 26345 27908 28339 33698
15.	Kata saku	2041 2340 8544 11723 11940 18351 21232 23525 29572 32252 32534 37422
16.	Lama beli	1791 2355 9251 13892 14988 20781 26289 26564 33174 35310 36841 41945
17.	Mana mami	1258 2456 8514 10655 11907 17868 21374 22473 28672 30455 31629 36235
18.	Pena napi	1165 1346 6920 8873 9897 15169 18965 19743 25526 27160 27384 32172
19.	Pola tapi	1533 1747 8431 10613 12185 18315 21712 21902 28834 30865 31064 37510
20.	Raja roma	1865 2435 9419 11840 12390 18546 24164 24815 31264 33702 34636 39383
21.	Sama nasi	3282 5884 13020 16922 18595 25382 30815 32175 39359 43483 44655 50333
22.	Tari satu	1566 1743 7503 10970 11713 17890 22400 23635 29513 31500 31716 36764
23.	Wana wisata	1382 2274 7599 9415 10545 15900 19144 19956 25401 29640 30375 36516 38402 38591 43596
24.	Kaya yoyo	1181 1504 7554 10295 11863 17298 20964 21889 28238 30443 31539 37448
25.	Zeni zona	1653 4658 10739 13453 14837 19843 23901 24947 30885 33472 34388 39455