Spoken Language Identification on Local Language using MFCC, Random Forest, KNN, and GMM

Vincentius Satria Wicaksana[1], Amalia Zahra, S.Kom, Ph. D.[2]
Computer Science Department, Bina Nusantara University, Jakarta, Indonesia

***Abstract*—Spoken language identification is a field of research that is already being done by many people. There are many techniques proposed for doing speech processing, such as Support Vector Machines, Gaussian Mixture Models, Decision Trees, and others. This paper will use the system using the Mel-Frequency Cepstral Coefficient (MFCC) features of speech input signal, use Random Forest (RF), Gaussian Mixture Model (GMM), and K-Nearest Neighbor (KNN) as a classifier, use the 3s, 10s, and 30s as scoring method, and use dataset that consists of Javanese, Sundanese, and Minang languages which are traditional languages from Indonesia. K-Nearest Neighbor has 98.88% of accuracy for 30s of speech and followed by Random Forest that has 95.55% of accuracy for 30s of speech, GMM has 82.24% of accuracy.**

Keywords—Gaussian mixture model; random forest; K-Nearest Neighbor; spoken language recognition; MFCC; GMM; KNN

I. Introduction

Indonesia is an archipelago in the Southeast Asia region. Indonesia consists of large islands and small islands spread from Sabang to Merauke, so that the Indonesian State is dubbed the Archipelago State. Indonesia is recorded as having 17.504 islands, therefore, the State of Indonesia has a variety of ethnicities, races, religions and cultures. Because of this diversity, Indonesia has a wide variety of languages, ranging from Javanese, Sundanese, Bahasa Batak, and many more. Therefore, some of regional languages in Indonesia are also extinct because the language is not widely used in the regions anymore. To prevent it from extinction, by collecting the dataset of regional languages to be studied, it can help to prevent extinction of regional languages, because when building a classification technique, a large scale of dataset is needed and by developing the SLI, the application can be used as a leading component of applications such as translators used to classify regional languages, which later can be used in speech-based information systems, speech-based translate, and others.

Referring to the problems above, the need for information technology solutions in the field of Spoken Language Identification is getting higher. Due to the Spoken Language Identification technology, Indonesian citizens who do not understand regional languages when visiting other areas or when tourists come to an area where residents do not understand Indonesian, can be helped by this technology.

In spoken language identification it takes several steps to identify a language, starting from the sound extraction such as MFCC [8] method to techniques for classifying language. Several techniques are used to classify languages, including deep neural networks [1], Gaussian mixture models [2][5][8], support vector machines [3], Random Forest [3], and others. Random Forest, KNN and GMM are technique that is quite widely used for classification, this technique has some parameter that can be tuned, which is very useful to increase accuracy. There has been a lot of research on spoken language identification, but no one has done research on spoken language identification that uses segmented speech. In this study, Random Forest, KNN and GMM will be used for classification techniques, the accuracy will be obtained from segmented speech in 3 seconds, 10 seconds, and 30 seconds. This study will examine spoken language identification using the techniques mentioned above and using the GMM technique which is often used in spoken language identification which is segmented at 3 seconds, 10 seconds, and 30 seconds as the baseline.

II. Literature Review

Spoken Language Identification (LID) [4] is a process for determining the identity of the language spoken. LID [4] is based on the linguistic properties of language obtained from the results of speech extraction. The performance of an LID system depends on the amount and reliability of information and how efficiently it is integrated into the system.

The sound structure of a language can be categorized into acoustic-phonetic, phonotactic, and prosodic. Acoustic-phonetic is one of the structures of the sound which is related to the analysis of the physical properties of the sound being spoken. While phonotactic is a sound structure related to the syllable structure of a language, for example Languages such as Dutch, English, and German allow a large number of consonants at the beginning and at the end of the syllable. In contrast, Maori, which is spoken in New Zealand, only allows syllables consisting of a vowel, two vowels, or a consonant plus a vowel. Prosody is a structure of sound related to rhythm, intonation, and stress of sound, for example, Mandarin has the same letter but has a different intonation, for example the word “ma” with high intonation means mother, while “ma” word with intonation drops later to ride means horse.

In [6], the authors discussed about spoken language identification using Shifted Delta Coefficient and Shifted Delta MLP as feature extraction and using Gaussian Mixture Model and Support vector Machine as classification technique.

In [2], the authors discussed about spoken language identification using Mel-frequency cepstral coefficients as feature extraction and using Gaussian Mixture Model as classification technique. In his research, to improve the performance of the Gaussian mixture model in his research, the total mixture was gradually added to get optimal results. In his research it was also explained that Tamil and Telugu languages have good performance by using mixture values between 128 to 512. Starting with 32 mixtures which produced low accuracy, namely 70% for Tamil and 85% for Telugu. Then the mixture components are increased little by little to get the desired results. When the mixture was increased to 128, the resulting accuracy was almost 100% for Tamil, while for Telugu, it got 100% accuracy. Then the mixture component is increased again to 512 which is the best point for the classification of the two languages, when the accuracy rate of both reaches 100%. If seen from the results above, it can be concluded that by gradually increasing the mixture component, it can improve performance in language identification. Despite of that, research that conducted by [7], discussed that by increasing the mixture component, the performance of the technique will increase, but the higher the mixture component will increase the computation cost or increase the time in computing.

In [3], the classification methods used are Support Vector Machine and Random Forest and for the feature extraction used are MFCC, LPC, and a combination of the two techniques, to find out which technique provides the best accuracy. To conduct an evaluation, [3] used the IIIT-H dataset which contains 5000 samples, consisting of 6 languages, namely Hindi, Telugu, Bengali, Marathi, Tamil, and Malayalam. Then 300 samples were taken randomly from the IIIT-H dataset using a 16kHz sound signal. The evaluation was carried out in 2 phases, the first phase was carried out using the MFCC feature with Support Vector Machine and Random Forest. Meanwhile, for the Random Forest technique, the resulting accuracy is 75.9% and 74.3% for the SVM technique. The second phase is carried out using the LPC feature with the same classification technique. The Random Forest technique used produces an accuracy of 61.5% and 67.16% for the SVM technique. In [3] is not stated why total trees of 300 has the best performance compared to the lower total of trees.

In [9], the study was conducted using a Gaussian mixture model as a technique for classifying languages which will be used to compare feature extraction. For feature extraction, MFCC, SDC, and a combination of both are used. In his research, the database used is the Arunachali Language Speech Database (ALS-DB), which consists of 6 languages, namely Adi, Apatani, Galo, Nyishi, Hindi and English. The experiment was carried out using the GMM with total of 1024 mixture component, MFCC features with 12 cepstral coefficient numbers.

Research conducted by Gupta et al. (2017) using Support Vector Machine and Random Forest. In his research, it is stated that by combining Mel Frequency Cepstral Coefficients (MFCCs) and Linear Predictive Coding (LPC) will increase the accuracy of the language identification. In [3], it is not explained why Gupta et al choose total trees of 300 compared to the lower total trees. In [3], only the total accuracy of the features extracted from the frame is explained, it is not explained how the accuracy from classification using 3s of speech, 10s of speech, and 30s of speech. Therefore, this paper will use MFCC and Random Forest to see the effect of the total trees on the accuracy and expanding the testing method by segment the duration of the test from frame to 3s speech, 10s speech, 30s speech and briefly discusses the performance of computation time when conducting model training using three traditional language from Indonesia. This paper will use KNN as a rarely use technique in language identification to see if its fits as classifier for spoken language identification and use GMM that widely use in language identification as baseline.

III. Proposed Method

A. Dataset

At the data collection stage, speech data collection will be collected both from the internet and from native speakers of the local language. In order to get the correct acoustic of the language, native speakers who are fluent in the regional language are needed. The dataset obtained will be divided into 2 datasets, the first is for the training dataset and the second is the test dataset. With a ratio of 70:15:15 , that is, 70% of the dataset will be used for training the dataset, 15% of the dataset will be used for validation dataset, and the other 15% of the dataset will be used for test dataset. The distribution of the dataset can be seen in Table I below.

The Javanese and Sundanese dataset will be obtained from *openslr* and Minang dataset language will be collected from *youtube* and recorded Speech.

B. Pre-Processing

After the data is collected, the Javanese and Sundanese language dataset will be sorted again. After that, the speech that obtained from YouTube will be processed again.

For Javanese and Sundanese dataset, each dataset will be combined into 80 minutes long. As for dataset that obtained from *youtube*, the first step is to remove the noise, song, background song, and unnecessary item from the recording. Same as Javanese and Sundanese dataset, Minang dataset will be combined or cut to achieve 80 minutes long of training dataset.

For test dataset, each language will have 20 minutes of speech data, the speech data will be divided into 3s, 10s, and 30s of speech data. After the recording is cut, the recorded file will be changed to wav format, because the compressed data produced by wav has a sound quality that is almost the same as the original sound. The sound will be resampled to 44.1 kHz and use a bit rate of 32 kbps.

TABLE I. TOTAL DURATION FOR EACH LANGUAGE

Language	Total Duration	Training Dataset Duration	Validation Dataset Duration	Testing Dataset Duration
Sundanese	200 Minutes	140 minutes	30 minutes	30 minutes
Minang	200 Minutes	140 minutes	30 minutes	30 minutes
Javanese	200 Minutes	140 minutes	30 minutes	30 minutes

C. Model Development

Every speech has its own characteristics, to get the characteristics of the speech, feature extraction will be executed. This study will use MFCC feature, the MFCC feature will be extracted from the pre-processed input signal. The MFCC feature will be presented using vector c, which a set of vector C has the value of C_1, C_2, C_3, C_4, C_5, $C_6 \ldots C_n$. In vector C, n represent the total coefficient will be extracted from the speech every frame. This study will use *python* and library provided from *librosa* to extract the MFCC feature. Total coefficient that will be used in the experiment is 13 and the total length of the frame is 25 milliseconds.

After extracting the features from the testing data and training data, a model development will be carried out. The model will be developed using random forest, Gaussian mixture model, and K-Nearest Neighbor. Random forest is a classification algorithm consisting of many decisions' trees. It uses bagging and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree. This study will use the random forest, Gaussian mixture model, and K-Nearest Neighbor and the library is provided by *sklearn*. Before using this model, the best parameter was determined for each method.

The Random Forest technique, there are several parameters will be tuned, such as n_estimator, criterion, max_depth, and max_sample_leaf. N_estimator is used to determine the total trees to be used, the random forest that used in [3] will be used as baseline. This total tress parameter will be used to compare the best n_estimator for this dataset. Criterion is used to measure the impurity of a node. Max_depth represents the depth of each tree in the forest. The deeper the tree, the more splits it has, and the more it captures more information about the data. The parameter for KNN will be tuned are the type of weight, total leaf size and number of neighbors and the last classifier is GMM that widely used for LID [2], [7], and [9].

D. Evaluation

In this study, the total percentage of accuracy will be measured. After the experimental process is complete, the evaluation results will be entered into a table for further observation. From table below, method column used to list the method that used, and the duration row is used to classify the average accuracy based on the duration of the speech.

The experiment flow can be seen from Fig. 1 below.

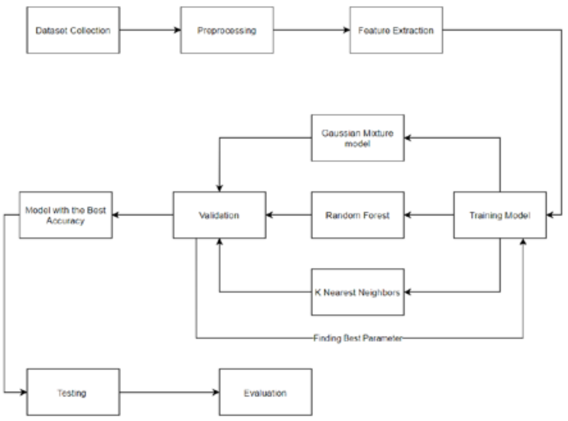

Fig. 1. Experiment Flow.

IV. Result and Discussion

The experiment is focused on comparing performance between three classifier techniques on three segmented duration. Classification was performed on three different language. Once the models are trained and the feature are extracted, the classifier was used to classify the dataset. The accuracy score of classification between three languages are reported in terms of percentage of accuracy. Tuning the min sample leaf parameter for random forest using validation dataset are recorded on Table II.

From Table II, the result shows us that there is no significant accuracy difference between each parameter, so min_sameple leaf will be set to the default value, the default value is 5.

From Table III, the result shows us that the higher max_depth number, the better accuray it has. The accuracy increases periodically as total max depth value increased and reach its peak at total max depth of 50, but from 50 to 100, there is no significant increase between 50 and 100, the score almost similar, so total max_depth of 50 will be used for this parameter.

From Table IV, the result shows us that there is no significant difference between criterion gini and entropy, but Entropy is more computationally heavy than gini, so it will increase the time computation, so gini will be used for this parameter.

TABLE II. FINDING BEST PARAMETER FOR MIN_SAMPLE_LEAF.

Parameters			**Accuracy**		
Min sample leaf	**N_estimator**	**N_jobs**	**3 s**	**10 s**	**30 s**
5	100	16	87.44%	90.18%	94.72%
10	100	16	87.11%	89.90%	94.44%
15	100	16	86.78%	89.35%	94.16%
25	100	16	86.47%	89.07%	93.88%

TABLE III. FINDING BEST PARAMETER FOR MAX_DEPTH

Parameters			**Accuracy**		
Max Depth	**N_estimator**	**N_jobs**	**3 s**	**10 s**	**30 s**
10	100	16	83.47%	83.99%	84.44%
15	100	16	85.16%	86.11%	88.61%
25	100	16	86.44%	88.25%	92.49%
50	**100**	**16**	**88.05%**	**91.29%**	**96.11%**
100	100	16	87.88%	91.57%	96.38%

TABLE IV. FINDING BEST PARAMETER FOR CRITERION

Parameters			**Accuracy**		
Criterion	**N_estimator**	**N_jobs**	**3 s**	**10 s**	**30 s**
Gini	100	16	88.00%	91.48%	96.38%
Entropy	100	16	87.78%	90.83%	96.11%

From Table V, the result shows us that there is no significant difference between 100 to 250 totals of trees, so total trees of 100 will be used in this experiment, gini as criterion parameter, min_sample_leaf of 5, and max_depth of 50 for this technique; fFor GMM, the search of the best parameter for number and mixture and covariance type.

In Table VI, full covariance type has the best score rather than the other covariance type. Next, the best parameter for GMM total mixture was tuned. [2] GMM accuracy was used as baseline.

In Table VII, by increasing total number of mixtures, the accuracy increased that is stated in [2]. Despite of that, there is a significant decrease from 128 to 256. In order to confirm that there are no errors in the code, the test was run 3 times and still has the same score, so the test is stopped at 256 and is decided that the total mixture of 128 has the best score for this dataset. Next, KNN is widely used on machine learning, but it is rarely used in language identification, so in this paper, KNN technique was used to determine if KNN is suitable for language identification or not. There are several parameters that will be used, such as K, weight, and size of leaf.

TABLE V. FINDING BEST PARAMETER FOR N_ESTIMATOR

Parameters				**Accuracy**		
N_estimator	**Min_sample_leaf**	**Max Depth**	**criterion**	**3 s**	**10 s**	**30 s**
100	5	50	Gini	87.52 %	90.92 %	95.27 %
150	5	50	Gini	87.63 %	90.27 %	94.72 %
250	5	50	Gini	87.62 %	90.09 %	94.72 %

TABLE VI. FINDING BEST PARAMETER FOR COVARIANCE TYPE

Parameters			**Accuracy**		
Covariance Type	**GMM Mixture**	**N_jobs**	**3 s**	**10 s**	**30 s**
Full	128	16	71.77%	77.77%	82.77%
Tied	128	16	61.86%	66.20%	70.00%
Diag	128	16	64.33%	70.92%	80.27%
spherical	128	16	62.19%	64.90%	64.16%

TABLE VII. FINDING BEST PARAMETER FOR NUMBER OF MIXTURE

Parameters			**Accuracy**		
Covariance Type	**GMM Mixture**	**N_jobs**	**3 s**	**10 s**	**30 s**
Full	16	16	45.45%	44.35%	57.77%
Full	32	16	56.25%	53.05%	63.88%
Full	64	16	75.61%	79.72%	75.55%
Full	**128**	**16**	**76.55%**	**80.09%**	**81.38%**
Full	256	16	46.27%	45.09%	48.33%

From Tables VIII and IX, there is not any significant difference from each parameter, each parameter has similar score, so the default value will be used for type of weight from the library which is uniform and total leaf of 20 because total leaf of 20 has the best better accuracy than 30 and 40, and it has less computation time. Next, the best parameter for K will be tuned for this dataset; the result can be seen from Table X.

From the table above, it can be concluded that by increasing the total of K, it will increase the score. Total k of 10 and 20 has the best accuracy, there is a slight difference between two of them, but the computation time and complexity must be considered, because by increasing the total K, the computation time and complexity will be increased. So, for this technique, total K of 10 will be used to get the accuracy from testing dataset. After getting the best parameter for each technique, each technique will be tested using dataset to get the accuracy result for each technique with tuned parameter.

In Table XI represents performance of feature with different classifier. The KNN classifier gives the highest score from 3 second of speech until 30 second of speech. RF gives almost similar score to KNN. GMM gives the lowest score in language identification from 3 sec, 10 sec, and 30 sec.

TABLE VIII. FINDING BEST PARAMETER FOR WEIGHT

Parameters			**Accuracy**		
Type of Weight	**K**	**N_jobs**	**3 s**	**10 s**	**30 s**
uniform	5	16	86.91%	91.01%	96.11%
distance	5	16	86.63%	90.64%	95.83%

TABLE IX. FINDING BEST PARAMETER FOR SIZE OF LEAF

Parameters			**Accuracy**		
Size of Leaf	**K**	**N_jobs**	**3 s**	**10 s**	**30 s**
20	5	16	86.91%	91.01%	96.11%
30	5	16	88.91%	90.01%	94.11%
40	5	16	87.91%	92.01%	93.11%

TABLE X. FINDING BEST PARAMETER FOR K

Parameters				**Accuracy**		
Size of Leaf	**Type of Weight**	**Total K**	**N_jobs**	**3 s**	**10 s**	**30 s**
30	uniform	5	16	86.36%	90.74%	95.83%
30	uniform	10	16	88.18%	93.61%	98.88%
30	uniform	15	16	85.38%	88.51%	91.38%
30	uniform	20	16	87.83%	91.12%	96.11%

TABLE XI. MODELS ACCURACY

Technique	**Accuracy**		
	3 s	**10 s**	**30 s**
MFCC + KNN	88.19%	93.61%	98.88%
MFCC + GMM	72.35%	80.59%	82.24%
MFCC + RF	87.66%	90.64%	95.55%

V. Conclusion

In this paper, this paper compares the widely use technique in language identification which is GMM and rarely use technique, which is KNN, and another technique called random forest to see if its good in segmentation speech or not.

From Table XI, KNN has the highest accuracy in each segment, with a score of 88.19% for 3s, 93.61% for 10s, and 98.88% for 30s, then followed by RF which has an accuracy score of 87.66% for 3s, 90.64% for 10s, and 95.55% for 30s. And GMM has the lowest score for each segmentation. However, when doing training and testing model for each technique, KNN use longer computation time when compared to Random Forest because KNN is called the lazy learner.

It can be concluded that KNN and RF is better than GMM and has the best accuracy for Javanese, Sundanese, and Minang.

Suggestion for the future research is to get more Minang dataset variation, such as high pitch speech, low pitch speech, and the other and using another feature extraction technique to see if there is a better feature extraction technique for KNN and RF.

Acknowledgment

Thanks to Amalia Zahra, S.Kom., Ph. D. from Bina Nusantara University for the useful discussions, guidance, and assistance in this research.

References

[1] Heracleous, P., Takai, K., Yasuda, K., Mohammad, Y., & Yoneyama, A. (2018). Comparative study on spoken language identification based on deep learning. *European Signal Processing Conference*, *2018-Septe*, 2265–2269. https://doi.org/10.23919/EUSIPCO.2018.8553347.

[2] Athiyaa, N., Jacob, G., Science, C., Anna, R., College, G., & Phil, M. (2019). Spoken Language Identification System using MFCC features and Gaussian Mixture Model for Tamil and Telugu Languages. 4243–4248.

[3] Gupta, M., Bharti, S. S., & Agarwal, S. (2017). Implicit language identification system based on random forest and support vector machine for speech. *2017 4th International Conference on Power, Control and Embedded Systems, ICPCES 2017*, *2017-Janua*, 1–6. https://doi.org/10.1109/ICPCES.2017.8117624.

[4] Lee, C. H. (2008). Principles of Spoken Language Recognition. *Springer Handbooks*, 785–796. https://doi.org/10.1007/978-3-540-49127-9_39.

[5] Chellappa, R., Veeraraghavan, A., Ramanathan, N., Yam, C.-Y., Nixon, M. S., Elgammal, A., … Reynolds, D. (2009). Gaussian Mixture Models. Encyclopedia of Biometrics, 659–663. doi:10.1007/978-0-387-73003-5_196.

[6] Wang, H., Leung, C. C., Lee, T., Ma, B., & Li, H. (2013). Shifted-delta MLP features for spoken language recognition. *IEEE Signal Processing Letters*, *20*(1), 15–18. https://doi.org/10.1109/LSP.2012.2227312.

[7] Kumar, A., Hemani, H., Sakthivel, N., & Chaturvedi, S. (2015). Effective preprocessing of speech and acoustic features extraction for spoken language identification. *2015 International Conference on.*

[8] *Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials, ICSTM 2015 - Proceedings*, *May*, 81–88. https://doi.org/10.1109/ICSTM.2015.7225394.

[9] Sarmah, K., & Bhattacharjee, U. (2014). GMM based Language Identification using MFCC and SDC Features. *International Journal of Computer Applications*, *85*(5), 36–42. https://doi.org/10.5120/14840-3103.