World Congress of International Fuzzy Systems Association 2011 and Asia Fuzzy Systems Society International Conference 2011, Surabaya-Bali, Indonesia, 21-25 June 2011, ISBN: 978-602-99359-0-5

Fuzzy Hidden Markov Models For Indonesian Speech Classification

*Intan Nurma Yulita
Telkom Institute of Technology
intanurma@gmail.com

The Houw Liong
Telkom Institute of Technology
houwthee@yahoo.co.id

Adiwijaya
Telkom Institute of Technology
adw@ittelkom.ac.id

Abstract: Indonesia has a lot of tribe, so that there are a lot of dialects. Speech classification is difficult if the database uses speech signals from various people who have different characteristics because of gender and dialect. The different characteristics will influence frequency, intonation, amplitude, and period of the speech. It makes the system must be trained for the various templates reference of speech signal. Therefore, this study has been developed for Indonesian speech classification. This study designs the solution of the different characteristics for Indonesian speech classification. The solution combines Fuzzy on Hidden Markov Models. The new design of fuzzy Hidden Markov Models will be proposed in this study. The models will consist of Fuzzy C-Means Clustering which will be designed to substitute the vector quantization process and a new forward and backward method to handle the membership degree of data. The result shows FHMM is better than HMM and the improvement was 3.33 %.

Keywords: Fuzzy, Hidden Markov Models, Indonesian, Speech, Classification

1 INTRODUCTION

Over the past several decades, the speech classification technology has been much done. There are many approaches to Speech Classification for example template-based, knowledge based, and stochastic-based approaches [13]. The successful results were the hidden Markov model (HMM) [8]. Other results were Artificial Neural Network [11], Support Vector Machine [10], Fuzzy [12] and Clustering [14].

Speech classification is a "language-dependent" system. The application of classification in a language cannot be applied into another language because each language has a list of phonemes. A lot of studies have been carried out abroad, but it cannot be applied well in Indonesian. English speech recognition is the most speech recognition system and has been developed in references [2, 3, 7, 8, 9, 10, 11, 12, 14]. The number of studies which has done Indonesia speech classification is still few. They have done Indonesian speech classification based on speaker adaptation system [6], and developed the corpus of Indonesian Speech Classification [4].

Speech classification is difficult because speech has some unique characteristics. In different time, a same word has different form although has been spoken from same person. So speech classification is more difficult if the database uses speech signals from various people who have different characteristics because of gender and dialect. The different characteristics will influence frequency, intonation, amplitude, and period of the speech. It makes the system must be trained for the various templates reference of speech signal. Therefore, a study still needs to be conducted.

Hidden Markov Models is a common approach used to classify speech. However, a method is needed to develop a solution from the above problem, and for Indonesian Speech classification. This study designs it. The solution combines Fuzzy on Hidden Markov Models. Fuzzy handles variant forms of speech more properly than there is no fuzzy. If the number of variant is higher, then the area of each cluster of Fuzzy C-Means Clustering is wider. Actually some study has combined fuzzy on HMM [5, 7, 8, 11] but they were not designed to solve the different characteristics problem in speech dialect and for Indonesian Speech. The new design of fuzzy Hidden Markov Models is proposed in this study. The model consists of Fuzzy C-Means Clustering which is designed to substitute the vector quantization process and a new forward and backward method to handle the membership degree of data.

2 MATERIALS AND METHODS

2.1 Raw data obtaining

In this study, it was used the speech recognition data from Research and Development Center of Telkom. Data collection was conducted in a soundproof room (it means there is no noise in speech) and the number of involved speakers was 70 people. Experiments were performed on speech data set with various characteristics dialect and gender. In this data, some dialect of speaker tribes in Indonesia was used, they were Sundanese, Javanese, Batak, Betawi, Balinese but there was no information how much their proportion. The data set was divided into training data (80% of data set) and testing data (20% of data set). The speakers of training data and testing data were different because our speech classification was speaker independent system. Table 1 lists the used words for the training data.

Table1. Training data

Words	Sounds	Information
Balaysalasa	/balaysalasa/ and /baleysalasa/	101 files from male and female
Lubuklinggaw	/lubuklinggaw/ and //lubuklinggo/	101 files from male and female
Prabumulih	/prabumulih/ and	101 files from

	/prabumuleh/	male and female
Tanjungenim	/tanjungenim/ and / tanjungénim/	100 files from male and female
Tarempa	/tarempa/ and /tarémpa/	98 files from male and female

Table 1 shows the extremely different sounds of each word.

2.2 Preprocessing

The purpose of preprocessing is to make all signal inputs conform with the required specifications in the system [2]. The first step is centering, it aims at shifting the location of the discrete amplitude distribution and it makes its center locate the axis y = 0. Thus, centering makes the average amplitude of the signal to zero. The next step is normalization, the process to equalize the maximum amplitude of the sound signal. Normalization is done by dividing each discrete amplitude values with the maximum amplitude value.

2.3 Feature Extraction

This process aims at obtaining the characteristics of the voice signal. In this study, MFCC is implemented for feature extraction. It produces 24 parameter values. They are 12 Cepstral values and 12 first-order derivative value of these Cepstral. The output of this process is that every speech is divided into a number of frames and each frame will have 24 feature values.

2.4 Vector Quantization (VQ)

Basically, the output of feature extraction is shorter than the original signal. However, in order to process HMM, an observation sequence is needed [2]. The observation represents all variation of existing Cepstral. VQ is used for the formation of discrete symbols (codebook) from a series of observations of the HMM model for estimating the vector representation of the shorter term.

VQ process is divided into two stages: the formation of codebook and the codebook index determination. When constructing codebook, the input feature vector of the VQ is a whole variety of known voice signal. By using clustering algorithms, feature vector will be grouped into clusters. The cluster center is called codebook. After the codebook is constructed, the next step of VQ can be done by replacing a feature vector with one vector codebook that has the smallest Euclidean distance. The output of VQ is the input of Hidden Markov Models.

2.5 Hidden Markov Models (HMM)

HMM is a Markov chain that its output symbol describes the chances of output symbol transitions [3, 9]. Observations for each state are described separately by a probability function or density function (probability density function), which is defined as an opportunity to produce a transition between states. Unlike the observable Markov model (OMM), HMM consists of a series of double stochastic process that primarily process cannot be directly observable (hidden) but can only be observed through another set of stochastic processes that produce a range of observations.

2.5.1 Basic Element

HMM as a discrete observation symbol has the following elements [3, 9]:

1. HMM consists of N states, they are labeled by {1, 2,..N} and state to-t is given by q_t. N is tested parameter in this study.
2. Number of observation symbols (M). Observation symbol is the output being modeled.
   $V= \{V_1, \ldots, V_m\}$
3. Transition probability distribution from one state to another state (A)
   $A= \{a_{ij}\}$, $1 \le i, j \le N$
4. Observation probability distribution of k^{th} symbol in the j^{th} state (B)
   $B= \{b_j(V_k)\}$, $1 \le i \le N$, $i \le j \le N$
5. Initial state probability distribution π_i
   $\pi_i = P(q_1=i)$, $1 \le i \le N$

HMM requires specification of two model parameters N and M. A, B, and π are measured. HMM notations are usually written with λ (model) = (A, B, π)

2.5.2 Basic problem and solution

There are three basic problems in HMM to be solved, namely [3, 9]:

1. If a given observation $O= \{O_1, O_2, \ldots, O_T\}$ and model evaluation $\lambda = (A, B, \pi)$, how to calculate the efficient probability of observations series?
2. If a given observation $O= \{O_1, O_2, \ldots, O_T\}$ and model evaluation $\lambda = (A, B, \pi)$, how to choose the optimal states series that represent the observation?
3. How to set the parameters of the model evaluation $\lambda = (A, B, \pi)$ to maximize the probability $P(O|\lambda)$ value?

The solution to the problem above is [3, 9]:

1. Evaluation (Evaluation of opportunities)
   The used common method is to examine every possible sequence of N states along the T (the number of observations). It is not efficient. Another simpler procedure is forward and backward procedures.

   A. Forward procedure
   Forward variable ($\alpha_t(i)$) at t-time and i-state is defined by $\alpha_t(i)= P(O_1, O_2, \ldots, O_T, q_t=i|\lambda)$ The forward opportunities function can be solved for N-state and T-symbol inductively with the following steps:

   a) Initialization :
   $$\alpha_1(i) = \pi_1 b_i(O_1),\ 1 \le i \le N \tag{1}$$
   b) Induction :
   $$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} (\alpha_t(i))\alpha_{ij}\right] b_j(O_{t+1}) \tag{2}$$
   $1 \le (i,j) \le N$, $1 \le t \le T-1$
   c) Termination :
   $$P(O|\lambda) = \sum_{i=1}^{N} (\alpha_T(i)) \tag{3}$$

   Forward probability is calculated based on the Trellis diagram pattern. There are n points each time slot in the pattern. All possible sequence is combined to N states.

B. Backward procedure

Backward variable β_t (i) in time to t and i-state is defined by $\beta_t(i) = P(O_{t+1}, O_{t+2}, \ldots O_T, q_t=1|\lambda)$. Step backward procedure is as follows:

a) Initialization :

$$\beta_t(i) = 1,\ 1 \leq i \leq N \quad (4)$$

b) Induction :

$$\beta_t(j) = \sum_{i=1}^{N} \alpha_{ji} b_j(O_{t+1}) \beta_{t+1}(j) \quad (5)$$

$1 \leq (i,j) \leq N$, t=T-1,T-2, ….. 1

To obtain the state to the i[th] time t and the rows of observations at time t +1, then it is assumed that the possible j-state at time t +1, to obtain a transition from i to j, and rows of observation on the j-th state. Then it calculates the observation of the j-state.

C. Forward-backward procedure

The combination of forward and backward procedure can be used to obtain the values of P (O|λ) Opportunity in the state at t-time of the N state before time t-1 can be calculated with the function of the forward opportunities $\alpha_t(i)$. Backward probability function , is used to calculate the probability of observation symbol sequence that it is started from time t + 1 to T.

By mathematical calculation, using *a forward-backward* procedure is illustrated as the following formula:

$$P(O|\lambda) = \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) \alpha_{ij} \beta_{t+1}(j) = \sum_{i=1}^{N} \alpha_t(i) \beta_t(i) \quad (6)$$

2. Decoding

The second problem is looking for the hidden *state* sequence *(hidden)* for a sequence of generated observations from model. The solution is used to find the *optimal state* sequence. It is Viterbi algorithm *(dynamic programming)*. Viterbi algorithm maximizes the probability value P(Q| O, λ) , so it will produce the optimal *state* sequence. Based on the Bayes rule, mathematically it is expressed as this formula:

$$P(Q|O,\lambda) = \frac{P(Q,O|\lambda)}{P(O|\lambda)} \quad (7)$$

3. The third problem solution is to adjust the *(training)* parameters , based on certain *optimal* criterion. The usual method to solve this third problem is the Baum-Welch algorithm. This algorithm is an iterative method that works to find the values of local maximum of the probability function. This *training* process continues until a critical state is met. The model result should be better *training* than the previous model.

2.6 Fuzzy Hidden Markov Models (FHMM)

The proposed FHMM does not implement vector quantization. The substituted process is Fuzzy C-Means Clustering. Fuzzy C-Means Clustering has two functions. First, it obtains the codebook by Clustering processing, the codebook is a cluster center. Second, it changes the feature extraction output to be the data with membership degree for each cluster. The data is used to be the Fuzzy Hidden Markov Models input.

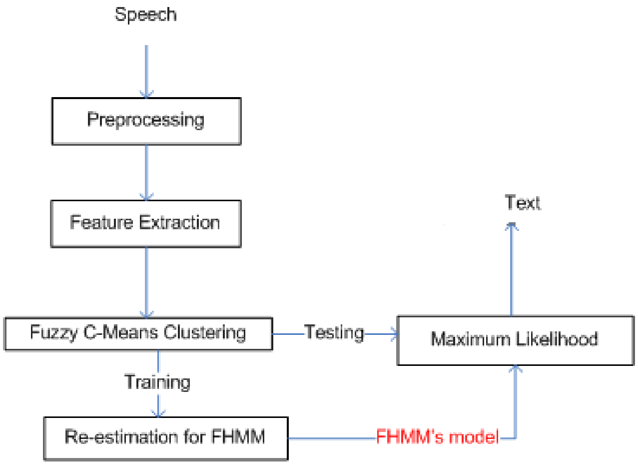

Figure 1. Speech classification using FHMM

From the block diagram above can be elaborated that the system is designed to have 2 ways (training and testing). Both ways have some same stage. They are preprocessing, feature extraction, and Fuzzy C-Means Clustering. The system input is speech. The speech is normalized. The normalized speech is extracted by feature extraction processing. The training of Fuzzy C-Means Clustering process is done to get codebook. After the codebook is constructed, the next step can be done by replacing a feature vector with a row of frame membership degree for each cluster. The testing of Fuzzy C-Means Clustering replaces a feature vector with a row of frame membership degree for each cluster. After Fuzzy C-Means Clustering, the training does re-estimation process for FHMM and the testing process decided the most similar reference model. The system output is text.

2.6.1 Fuzzy C Means Clustering

The steps of Fuzzy C-Means Clustering will be shown in the following steps [1]:

1. Initial data input, matrix X, with size nxm, (n = number of frames, m = number of features)
2. Determining the parameters:
   a) Number of clusters (k) : tested parameter
   b) Maximum iterations (t) : 1000
   c) The expected smallest error : 10^{-5}
   d) Iteration start : 1 (one)
   e) Power (w) : tested parameter

   The number of cluster indicates the variation of recognized sound. If the number of cluster is 16 then there are 16 variation of recognized sound. The power of Fuzzy C-Means Clustering indicates range of each cluster. If the power is 2 then the cluster range is wider than if the power is 1.3. It means if the power is 2 then membership degree of data is higher than the power is 1.3.

3. Generating random values from the matrix U which is a matrix number of frames, and the number of clusters, to make the matrix elements of the initial partition U.

Calculating the partition matrix (μ_{ik}):

$$\mu_{ik} = \frac{\mu_{ik}}{Q_j} \tag{8}$$

4. Calculating the k^{th} cluster center (V_{kj}) :

$$V_{kj} = \frac{\sum_{i=1}^{n}\left((\mu_{ik})^{w} X_{ij}\right)}{\sum_{i=1}^{n}(\mu_{ik})^{w}} \tag{9}$$

5. Calculating the objective function (P_t) at iteration t:

$$P_t = \sum_{i=1}^{N}\sum_{k=1}^{c}\left(\left[\sum_{j=1}^{m}\left(X_{ij} - V_{kj}\right)^2\right](\mu_{ik})^{w}\right) \tag{10}$$

6. Doing iteration and at each iteration the partition matrix(μ_{ik}) will be updated:

$$\mu_{ik} = \frac{\left[\sum_{j=1}^{m}\left(X_{ij} - V_{kj}\right)^2\right]^{\frac{-1}{w-1}}}{\sum_{k=1}^{c}\left[\sum_{j=1}^{m}\left(X_{ij} - V_{kj}\right)^2\right]^{\frac{-1}{w-1}}} \tag{11}$$

7. Checking the stop condition:
   1) If new objective function value less the same old objective function value is less than the expected error value, or more than the maximum t value iteration, ($|P_t - P_{t-1}| < \xi$) or (t>MaxIter), then stop
   2) Step 4 will be repeated if the condition has not stopped and t=t+1

Fuzzy C-Means Clustering is done to obtain the cluster center (codebook). After the codebook is constructed, the next step can be done by replacing a feature vector with a row membership degree of frame for each cluster.
After the codebook is obtained, then calculate membership degree of data for each cluster ($\ddot{O}_{xz}$) using the following equation:

$$\ddot{O}_{xz} = \frac{\left[\sum_{y=1}^{m}\left(X_{xy} - V_{zy}\right)^2\right]^{\frac{-1}{w-1}}}{\sum_{z=1}^{c}\left[\sum_{j=1}^{m}\left(X_{xy} - V_{zy}\right)^2\right]^{\frac{-1}{w-1}}} \tag{12}$$

Note:

a) x : number of frames of observation data
b) y : number of features
c) z : number of clusters

2.6.2 Fuzzy Forward-Backward

The difference between the HMM and the FHMM is for each observation HMM refers to one codebook value of one frame and while in FHMM, observation refers to a frame value but it has all the values in each codebook with different membership degree. Therefore, a new framework of forward and backward calculation needs to be conducted. In this sub-chapter, the other forward and backward calculation is also shown [5].

Initialization of forward calculation (t=1):

a. HMM :

$$\alpha_1(i) = \pi_1 b_i(O_1) \tag{13}$$

b. The proposed FHMM :

$$\dot{\alpha}_1(i) = \pi_1 b_i(\dot{O}_1) \tag{14}$$

c. Others FHMM :

$$\alpha_1(i) = \pi_1\left[\sum_{m=1}^{M} u(m,1) b_i(O_1)\right] \tag{15}$$

Induction of forward calculation (t=2, ...T):

a. HMM :

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N}\left(\alpha_t(i) a_{ij}\right)\right] b_j(O_{t+1}) \tag{16}$$

b. The proposed FHMM :

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N}\left(\alpha_t(i) a_{ij}\right)\right] \dot{b}_j(\dot{O}_{t+1}) \tag{17}$$

c. Others FHMM :

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N}\left(\alpha_t(i) a_{ij}\right)\right]\left[\sum_{m=1}^{M} u(m,t) b_j(m)\right] \tag{18}$$

Induction of backward calculation (t=T):

a. HMM :

$$\beta_t(j) = \sum_{j=1}^{N} a_{ji} b_j(O_{t+1}) B_{t+1}(j) \tag{19}$$

b. The proposed FHMM :

$$\beta_t(j) = \sum_{j=1}^{N} a_{ji} \dot{b}_j(\dot{O}_{t+1}) B_{t+1}(j) \tag{20}$$

c. Others FHMM :

$$\beta_t(j) = \left[\sum_{j=1}^{N} a_{ji} B_{t+1}(j)\right]\left[\sum_{m=1}^{M} u(m,t) b_j(m)\right] \tag{21}$$

Calculation of forward-backward:

a. HMM :

$$P(O \mid \lambda) = \sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_t(i) a_{ji} b_j(O_{t+1}) B_{t+1}(j) \tag{22}$$

b. The proposed FHMM :

$$P(O \mid \lambda) = \sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_t(i) a_{ji} \dot{b}_j(\dot{O}_{t+1}) B_{t+1}(j) \tag{23}$$

c. Others FHMM :

$$P(O \mid \lambda) = \sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_t(i) a_{ji} B_{t+1}(j)\left[\sum_{m=1}^{M} u(m,t) b_j(m)\right] \tag{24}$$

Note :

a. $$\dot{b}_i(\dot{O}_x) = \sum_{z=1}^{c} B_{zi} \ddot{O}_{xz} \tag{25}$$

   This formula means that the input data is observation data which has membership degree for each cluster, and the output data is the observation probability distribution of x^{th} symbol in the i^{th} state (B).

b. u(m,t)=similarity(cb(m),O_t) (26)

   cb (m) is a cluster center vector for index m.

c. Similarity measure m (represents the number of features)

Table 2. Similarity measure m

Cosine similarity (27)	$(x_i, x_j) = \frac{\sum_{k=1}^{m} X_{ik} X_{jk}}{\sqrt{\sum_{k=1}^{m}(X_{ik})^2 \sum_{k=1}^{m}(X_{jk})^2}}$
Manhattan distance (28)	$(x_i, x_j) = \sum_{k=1}^{m}\left\|X_{ik} - X_{jk}\right\|$
Euclidean distance (29)	$(x_i, x_j) = \sqrt{\sum_{k=1}^{m}\left\|X_{ik} - X_{jk}\right\|^2}$

d. The four formulas of the proposed forward and backward calculation are changed because every value $b_j(O_t)$ refers to all codebook with different degrees of membership.

3 EXPERIMENTAL RESULTS

3.1 Compare HMM and FHMM if the number was altered

The purpose of experiment was to obtain the optimal number of cluster. The static variables were the number of states and the power (w). In this experiment, the number of state was 7(seven) and the power (w) was 1.1.

Table 3. If the number of cluster was altered

Method	The number of cluster	
	16	32
HMM	66.67%	80 %
FHMM	84.17%	88.33%

Table 3 shows the accuracies of HMM and FHMM if the number of cluster was altered. If w increased then FHMM and HMM accuracies increased. The optimal number of cluster was 32. It means that the system required 32 variant of recognized sound to obtain a good accuracy. The experiment did not try if the number of cluster was 64 because since this study had only five recognized words, all words had few phonemes. If the number of cluster was 64, it would cause the overspecialization system.

3.2 Compare FHMM for each power (w)

The purpose of experiment was to obtain the optimal power (w) of FHMM. The static variables were the number of states and the number of cluster. In this experiment, the number of state was 7 and the number of cluster was 32. The number of cluster was 32 because it was the optimal number which was obtained from experiment of table 3.

Table 4. If the power (w) was altered

w	Accuracy
1.05	92.5 %
1.1	88. 33 %
1.3	83. 33 %
1.5	65 %
1.7	46. 67 %

From table 4 shows FHMM accuracy if power (w) was ranging from 1.05-1.7. The optimal power (w) was 1.05 and if w increased then FHMM accuracies decreased. The explanation of the result will be shown in the following figure:

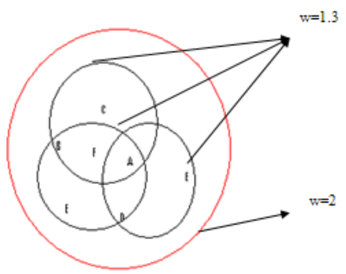

Figure 2. The influence of w

If the power (w) was 1.3, each data had different degrees of membership for each cluster. Otherwise, if the power (w) was 2, three clusters have the same region and each data has the same degrees of membership for each cluster. It means that there is no different among observation data and the system will only recognize one label.

3.3 Compare FHMM for each state

The purpose of experiment was to obtain the optimal number of state. The static variables were the number of cluster and the power (w). In this experiment, the number cluster was 32 and the power was 1.05. The parameter values were the optimal values which were obtained from experiment of table 3 and 4.

Table 5. If the number of state was altered

Method	The number of state					
	5	6	7	8	9	10
HMM	80 %	86.67%	80 %	86.67%	89.17%	85.83%
FHMM	90 %	90 %	92.5%	90 %	91.6 %	91.67%

From table 5, the optimal number of state of HMM was 9 and the optimal number of state of FHMM was 7.The accuracy was not influenced the number of states because if the number of state was increased, the accuracy sometimes increased and decreased.

3.4 Compare HMM and FHMM

The purpose of experiment was to compare HMM and FHMM if they had the optimal condition (the best accuracy). The optimal condition of HMM was if the number of cluster was 32 and the number of state was 9. The optimal condition of FHMM was if the number of cluster was 3, the power (w) was 1.05, and the number of state was 7.

Table 6. HMM and FHMM

Method	Accuracy
HMM	89.17%
FHMM	92.50 %

From table 6, FHMM was better than HMM. FHMM could improve HMM accuracy and its improvement was 3.3333 %.

4 CONCLUSSION AND RECOMMENDATION

4.1 Conclussion

From the analysis of the performance of FHMM by using the data in this study, it can be concluded that the optimal condition of FHMM to obtain a good accuracy in this study are the number of cluster is 32, the number of state is 7, and the power (w) is 1.05. With this optimal condition, the FHMM's accuracy is 92.50 % and it is better than the HMM's accuracy, the improvement is 3.33 %.

4.2 Recommendations for future works

Since our method is an effective way to Indonesian Speech Classification, it is strongly recommend the use of FHMM for a bigger database, and the implementation of more efficient time complexity on FHMM. The proposed method needs longer time than HMM. Other recommendation is the use of bigger frequency on FHMM.

REFERENCE

[1] Book: Sri Kusumadewi, and Hari Purnomo: Aplikasi Logika Fuzzy untuk Pendukung Keputusan, Penerbit Graha Ilmu, pp. 84-85, 2004.

[2] Book: B.H. Juang and Lawrence R. Rabiner: Fundamentals of Speech Recognition, Prentice-Hall International, Inc, 1993.

[3] Journal: B. H. Juang; L. R. Rabiner, Hidden Markov Models for Speech Recognition, Technometrics, Vol. 33, No. 3., pp. 251-272. 1991.

[4] Journal: Dessi Puji Lestari,Koji Iwano, Sadaoki Furui: A Larger Vocabulary Continuous Speech Recognition System for Indonesian Languange, 15th Indonesian Scientific Conference in Japan Proceedings, ISSN: 1881-4034, 2006.

[5] Journal: Harun Uguz , Ali Ozturk, Rıdvan Saracoglu, and Ahmet Arslan: A Biomedical System Based on Fuzzy Discrete Hidden Markov Model for The Diagnosis of The Brain Diseases, Expert Systems With Applications 35 1104–1114, 2008.

[6] Journal: Hammam Riza, and Oskar Riandi: Toward Asian Speech Translation System: Developing Speech Recognition and Machine Translation for Indonesian Language, International Joint Conference on Natural Language Processing, 2008.

[7] Journal: Jia Zeng, and Zhi Qiang Liu: Interval Type-2 Fuzzy Hidden Markov Models, Proceedings of International Conference on Fuzzy Systems vol.2 pp.123 - 1128 2004.

[8] Journal: Jia Zeng And Zhi-Qiang Liu: Type-2 Fuzzy Hidden Markov Models to Phoneme Recognition, Proceedings of the 17th International Conference on Pattern Recognition, 2004.

[9] Journal: Lawrence R. Rabiner: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Proceedings of the IEEE, Vol.77, No.2, 1989.

[10] Journal: Lei Chen, Sule Gunduz, and M. Tamer Ozsu: Mixed Type Audio Classification with Support Vector Machine, Proceedings of the IEEE International Conference on Multimedia and Expo, 2006.

[11] Journal: Patricia Melin, Jerica Urias, Daniel Solano, et all: Voice Recognition with Neural Networks, Type-2 Fuzzy Logic and Genetic Algorithms, Engineering Letters, 13:2, 2006.

[12] Journal: Ramin Halavati, Saeed Bagheri Shouraki, Mahsa Eshraghi, Milad Alemzadeh: A Novel Fuzzy Approach to Speech Processing, 5th Hybrid Intelligent Systems Conference, 2004.

[13] Journal: Sinout D. Shenouda, Fayez W. zaki, Amr Goneid: Hybrid Fuzzy HMm System for Arabic Connectionist Speech Recognition, Proceedings of the 5th WSEAS International Conference on Signal Processing, robotics and Automation, pp 64-69, 2006.

[14] Journal: Stephen E. Levinson, Lawrence R. Rabiner, Aaron E. Rosenberg, and Jay G. Wilpon: Interactive Clustering Techniques for Selecting Speaker-Independent Reference Templates For Isolated Word Recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. Assp-27, 1979.