Nearest Centroid Neighbour, An Alternative in the Speech Recognition for the execution from a Mobile Robot Simulator
(Applied to the Imbalanced Training Sample Problem)

E. RANGEL [1] and O. GARCÍA [1]

[1] Laboratorio de Reconocimiento de Patrones, Instituto Tecnológico de Toluca, Ave. Tecnológico s/n, 52140 Metepec, México.

erangel_lugo@hotmail.com
meza37@hotmail.com

Abstract

In this paper we show a knowledge-based system by speech recognition in the input data, it allows to execute a mobile robot simulator of Ayllu´s routines. Commonly for the classification phase in speech recognition are used the supervised neural networks (the BackPropagation Algorithm [BPA] which has been extensively used), and have shown a best performance. However when the knowledge-based system uses a training sample with imbalance, the neural networks (BPA) are not a good choice. We present an alternative for the classification using k-NCN rule in the speech recognition applied to the imbalanced training sample problem. Novel definition of a weighted function concerning the nearest centroid neighborhood is also introduced.

Keywords: Nearest Centroid Neighbour rule, Neural Network, Speech Recognition, Ayllu´s Task.

1. Introduction

The expert systems commonly need a knowledge base module and inference engine [**1**]. The knowledge base has questions and answers concerning the solution for the assigned problem. The inference engine is used for the discrimination from the information included in the knowledge base, also it is in charge of the decisions performance by the system. However, in the supervised methods, a knowledge-based system, which has learning, uses a training sample [**2**]. From the training sample depends the learning phase, thus the supervised method will learn from it. The neural networks (BPA) [**3**], in the learning phase, use the training sample to obtain the generalized weights. In the classification phase, the weights of the neural networks are used and it is not necessary the use of the training sample any more [**4**]. However, in the nearest neighbor rule [**5**], the training sample is used in both phases: Learning and Classification.

In the mobile robotics area, commonly before practicing in real time any experiment, it is used a simulator where all the moves and roads to executing with the robot are studied [**6**]. In this simulator the programming errors are observed, and can be corrected later. Some well-known mobile robots are the Pioneer's collection of ActivMedia [**7**]. There are many programming languages for a mobile robot of ActivMedia [**8**], some of them are Saphira, Java, Ayllu, and C++. The routines for Shapira, C++, and Java must always be programmed and it depends on the routine to be executed, which might be difficult for programming. The Ayllu language in its user's manual [**9**] has some routine, command and tasks included that can be adapted to the necessity. These routines are very easy for programming or for adapting, allowing to execute the command or tasks without any problem, or damage suffered by the robot. The best-known Ayllu´s routines are four and allow you to execute different task and commands. The routines are named: StraightLine, Wanderer, Goto and Getto. In speech recognition area, it is possible to understand spoken messages by the human voice through from the computer. That is possible, when some techniques concerning language structure, automatic learning, Hidden Markov Models, semantic analysis, analogical frequency analysis and digital frequency analysis are used [**10**]. There has never existed a speech recognition system that understands all the messages input introduced by the human. However, there are many systems, which allow you to work with an acceptable error, even with the very restricted semantic [**11**]. The speech recognition systems learn with a training sample that has frames from voice and text captured [**12**]. The frame is a feature vector that has a smaller part from digital frequency. This frame is labeled in this phase. The classification phase in the speech recognition systems, is also named recognition phase. In this phase, the system allows the identification of human voice; to be able to execute the commands introduced by the microphone. In areas from the speech recognition and mobile robotics; the neural networks (BPA) have been very studied. In speech recognition, they have been commonly used for the classification of frames [**13**] and in mobile robotics to execute the road commands in real time when environment images are used [**14**, **15**]. In those applications there has never been studied the imbalanced training sample problem.

The problem of imbalanced training data in supervised machine learning methods is receiving growing attention. Imbalanced data means that one class is much more represented than the others in the training sample. It has been observed that this situation, which arises in several practical domains, may produce an important deterioration of the classification accuracy, in particular with patterns belonging to the less represented classes [**16**, **17**]. The imbalanced training sample problem has been studied by Bruzzone and Serpico [**18**] in neural network area. They show that some neural networks in the learning phase cannot obtain the best performance in the generalized weights for the minority class. Then in the classification phase, does not exist enough information in the weights for new prototypes of the minority class. However R. Barandela, et al. [**17**] have proposed to use the nearest neighbor with a weighted distance function, as an alternative solution in the imbalanced training sample problem.

In the classification phase, we show the comparison of results obtained from Nearest Centroid Neighbor and the Neural Network (BPA).

2. Experimented Strategies

The NN rule is a well-known supervised non-parametric classifier that combines its conceptual simplicity and an asymptotic error rate conveniently bounded in terms of the optimal Bayes error [**17**]. In its classical manifestation, given a set of n previously labeled prototypes or training sample (*TS*), this classifier assigns any given instance to the class indicated by the label of the closet prototype in the *TS*. More generally, the k-NN rule maps any instance to the pattern class most frequently represented among its k closets neighbors. Nevertheless, the NN classifier also suffers from certain drawbacks. The performance of these rules, as with any non-parametric method, is extremely sensitive to incorrectness or imperfections in the training sample. On the other hand, its applicability to real-time problems, with a large set of training patterns of high dimensionality, can become prohibitive because of the immense computational load required for searching the nearest neighbor of each new pattern in the training sample. As a technique for internally biasing the discrimination procedure, we have experimented with a modification of the Euclidean metric that can be regarded as a weighted distance function. With this modification, when classification of new pattern Y is attempted, and in the search through the training sample of its nearest neighbor, the following quantity must be computed for each training instance:

$$d_W\ (Y, x_0) = (n_i / n)^{1/m}\ d_E\ (Y, x_0) \qquad (2.1)$$

Where:

x_0 is a training pattern representing class i.
n_i is the number of training patterns from class i.
n is the training sample size.
m is the dimensionality of the features space.
$d_E(.)$ is the Euclidean metric.

The weighted distance function criteria, is usually employed with 1-NN and k-NN rules [**17**] to improve the classification in the imbalanced training sample problem. In this work, we present a new proposal using a weighted distance function into the classification with k-NCN rule. The following sections show some related techniques used.

2.1.- The Weighted k-NCN Classification (New proposal)

The nearest centroid neighborhood concept is basically focused on the idea that the neighborhood of a point is subject to two complementary constraints. First, by the *distance criterion*, the k neighbors of a sample, say p, are as near to it as possible. The distance criterion commonly used is the Euclidean metric. Second, by the *symmetry criterion,* their centroid is also as close to p as possible. Note that the traditional nearest neighborhood used by the k-NN rules as well as the *LVQ* algorithms [**19**] just takes the first property into account, and so the nearest codebook vectors may not be symmetrically distributed around the sample p in the input space. Let p be a point whose k nearest centroid neighbors (*NCN*) should be found in a set of samples (training set), $X = \{m_1, \ldots, m_n\}$. These k neighbors can be searched for through an iterative procedures [**20**] in the following way:

1. The first *NCN* to p correspond to its *NN*, say q_1.
2. The i'th neighbor, q_i , $i \geq 2$, is such that the centroid of this and all previously selected neighbors, say $q_1, \ldots, q_{i-1}$ is the closets to p.

This procedure gives rise to a kind of neighborhood in which spatial distribution of neighbors is taken into account because of the centroid criterion. Furthermore, proximity of the k-NCNs to the sample is guaranteed because of the incremental nature of the way in wich they are obtained from the first NN. Using this neighborhood, it is possible to define the so-called k-NCN classification rule [**21**] as follows:

1. Find the k-NCNs to p, say $X^P = \{m^P_1, \ldots, m^P_k\}$, where $k \leq n$.
2. Assign to p the class with a majority of votes among its k-NCNs in the set X^P (resolve ties randomly)

The design of the Weighted k-NCN Classification, consists of the modification of the distance criterion (do not modify centroid criterion), changing only the Euclidean distance function criterion used commonly by a weighted distance criterion above-mentioned ($d_W\ (Y, x_0) = (n_i / n)^{1/m}\ d_E\ (Y,x_0)$). The assignment from weights for the n_i value corresponds to the next prototype in the training sample, which become centroid part. The procedure Weighted k-NCN Classification can be written as follows:

1. Let TS_i the i^{th} prototype of the training sample.
2. Let f the features number.
3. Let C_i the i^{th} centroid obtained for each prototype.
4. Assign to the k value.
5. Let A the integer number, which it represents the current k-NCN
6. Put $A = 0$.
7. For each new pattern X:
   - Find the 1-NN (using the weighted distance function) which must be assigned as the 1-NCN.
     - Put A=1
     - For finding the next weighted k-NCNs in iterative form as follows:
       a) Put $j = A$;
       b) Let q_j the jth-NCN Weighted.
       c) Calcule the Centroids among the NCN q_j (in this time) with each prototype TS_i.. Calcule this Centroid how follows:
       d) For $h = 1$ to f do
          If $j{=}{=}1$ $C_{i,h} = (q_{1,h} + TS_{i,h}) / 2$
          Else $C_{i,h} = (q_{1,h} + ... + q_{j,h} + TS_{i,h}) / (j+1)$
          End-If End-For
       e) Calcule the distances among the new pattern X and each Centroids C_i. Use the weighted distance function criterion. The weighted for the value n_i, should be the corresponding to the class of TS_i.
       f) Let TS_c the prototype with minor distance. Where c is the number of the prototype NCN.
       g) Put $A = A + 1;$
       h) Put $q_A = TS_c$.
       i) If $A \geq k$ *break*
   - Once finding the k-NCN´s, X is assigned to the majority class among those k-NCN. If a majority class does not exist among those weighted k-NCNs, the results are decided randomly.

The idea with the Weighted k-NCN is to improve the classification in the minority class, using centroid criterion and Weighted distance function.

2.2.- Frames Selection Procedure in the Speech Recognition

This technique was used for obtaining the features selection of the frame (In the system module related with speech recognition). The process for the selection from frames [**10**] can be written as follows:

1. Convert the analogical frequency to digital frequency.
2. Choose a smaller digital frequency frame (approximately 30 ms)
3. Save the frame with the features, in the range from 30 ms
4. Forward 10 ms and repeat the process (2-4) until to have obtained the features from all digital frequency
5. When (Step 4 has Finalized) do:
   If (Learning Phase) Then Call human expert,
   for assigning the label to the frame
   Else The Classification rule assigns the label
   to the frame
   End.

2.3.- The Neural Network (BAP) used

The best results of the neural network (BAP) [**3**] were obtained when we have only used three layers: Input Layer with 20 nodes, Output Layer with 5 nodes and the Hidden-Layer with 21 nodes. The reason of learning used was:

$$\Delta w_n = -\eta_n \left| E_w''(n) \right|^{-1} E_w'(n) \qquad (2.3.1)$$

The process iterative for the learning phase of backpropagation algorithm can be written as follows:

1. For each vector x_k in the training sample.
2. Compute the lineal functions of the network output and the subtraction: $\varepsilon_k = (d_k - s_k)$.
3. The network weights are updated by event (for each prototype x_k):

$$W_{kj}(t+1) = W_{kj}(t) + \Delta_p W_{kj}(t) \qquad (2.3.2)$$

Were:

$$\Delta_p W_{kj} = \eta (y_{pk} - o_{pk}) f_k(neta_{pk}) i_{pj} \qquad (2.3.3)$$

Then, the updating from weights can be written as follows:

$$W_{kj}(t+1) = W_{kj}(t) + \eta (y_{pk} - o_{pk}) i_{pj} \qquad (2.3.4)$$

4. Repeat the steps 1-3 for all the prototypes (P), which are included in the training sample.
5. If the squared error means

$$E = \frac{1}{2P} \sum_{k=1}^{P} \varepsilon_k^2 \qquad (2.3.5)$$

has a small value and acceptable, the learning phase is finished. Otherwise, repeat from the Step 1 with all the prototypes (x_k) of the training sample.

The Input Layer has 20 nodes concerning to the features that in the training samples are contained (To see Table 1). However, the Output Layer has 5 nodes, thus in the training sample five classes are contained (To see Table 1). Finally, the Hidden-Layer experiments were carrying out using some variants:

- One Hidden-Layers with 21 nodes
- Two Hidden-Layers with 21 nodes
- Three Hidden-Layers with 21 nodes
- One Hidden-Layers with 100 nodes
- Two Hidden-Layers with 100 nodes

On the other hand, for the classification phase the weights obtained in learning phase are used.

TABLE 1. CHARACTERIZATION OF THE ORIGINAL DATASET.

VOICES-ROUTINES			
# of Features		20	
# of Classes		5	
# Global of prototypes (Frames)		880	
Class	# prototypes by class	Routine to execute	
		English	Spanish
0	200	StraightLine	LineaRecta
1	200	Wanderer	Rotacion
2	200	Goto	Ir_a
3	80	Getto	Resuelve_Ir_a
4	200	Empty	Line_In

3. Experiments and Results

First, it was necessary to carry out the implementation from the mobile robot simulator. This application was developed in Borland C++ Builder programming´s language. The mobile robot simulator allows the execution from the best known four Ayllu´s routines:

StraightLine: The Robot Walks while the road has not obstruction, if it detects some obstacle, it goes backward.

Wanderer: The Robot Walks while the road has not obstruction, if it detects some obstacle it executes a rotation. Also this routine allows you to carry out a Wall Pursuit (Following - Wall).

Goto: This routine is used for arriving at the coordinate (X,Y) defined by the user, if some obstacle is detected in the road, the robot stops or it executes the StraightLine routine.

Getto: This routine is used for arriving at the coordinate (X,Y) defined by the user, when there are some obstacles in the road. The robot solves doing a rotation and calculating the coordinate again to be able to arrive at (X,Y).

The most interesting Ayllu´s routine is Getto; thus it allows solving obstruction in the road. If the coordinate (X,Y) is situated in a different room, where the robot is present, Getto can be able to arrive very easily and without any problems, or damage suffered by the robot.

The knowledge-based system experiments were carried out in two phases:

1. *The Learning phase*
2. *The Classification phase*

The knowledge-based system proposal in this work uses an internal expert system for the language recognition: English and Spanish. The expert system allows you to identify the words text to be well understood and to avoid confusions from assigned label in the learning and classification phases.

3.1.- The Learning Phase for The Knowledge-Based System Model

In *Figure 1* the process used in the learning phase it is shown. This process consists of the incorporation of prototypes in the training sample and also in the knowledge base. In the training sample, we have incorporated the frames obtained from digital frequency. These frames were obtained with the procedure above-mentioned (section 2.2). However, in the knowledge base, there have been added some text words captured by the keyboard concerning the voice captured by the microphone. Then the knowledge base is a small dictionary of question and answers that has routine words in the languages English and Spanish. For the incorporation of the prototype (frame) in the training sample, first the human voice by the microphone was captured; immediately it was saved in "wave" file extension. Then the wave file was opened to be applied the Frames Selection Procedure. Once labeled the frame, it was incorporated in the training sample and the text word spoken was added in the dictionary of the knowledge based, where the answers and questions are related with the frame features. For the incorporation of the other prototypes it was necessary to repeat this process. The Learning phase was finalized when in the training sample many prototypes of different strata were added. For the different strata the voices from men and women are both considered. In Table 1 the characteristics of the training sample resultant from the learning phase are shown. As you can observe, there is imbalance in the training sample for the most important routine: Getto. The file of the training sample was named: VOICES-ROUTINES. However, in the Table 2, another characterization from the training sample: VOICES-ROUTINES (when the cross validation has been used) is shown. On the other hand, the neural network (BAP) used in this work, did use the training sample VOICES-ROUTINES in its learning phase to obtain the generalized weights that will be used in the classification phase.

TABLE 2. CHARACTERIZATION OF THE DATASET, WHEN THE CROSS VALIDATION HAS BEEN EMPLOYED.

<table>
<tr><td colspan="5">VOICES-ROUTINES</td></tr>
<tr><td colspan="3"># of Features</td><td colspan="2">20</td></tr>
<tr><td colspan="3"># of Classes</td><td colspan="2">5</td></tr>
<tr><td colspan="3"># Global of prototypes (Frames)</td><td colspan="2">880</td></tr>
<tr><td colspan="5"># prototypes by class</td></tr>
<tr><td colspan="5">Training Sample</td></tr>
<tr><td>Class 0</td><td>Class 1</td><td>Class 2</td><td>Class 3</td><td>Class 4</td></tr>
<tr><td>160</td><td>160</td><td>160</td><td>64</td><td>160</td></tr>
<tr><td colspan="5">Test Sample</td></tr>
<tr><td>Class 0</td><td>Class 1</td><td>Class 2</td><td>Class 3</td><td>Class 4</td></tr>
<tr><td>40</td><td>40</td><td>40</td><td>16</td><td>40</td></tr>
</table>

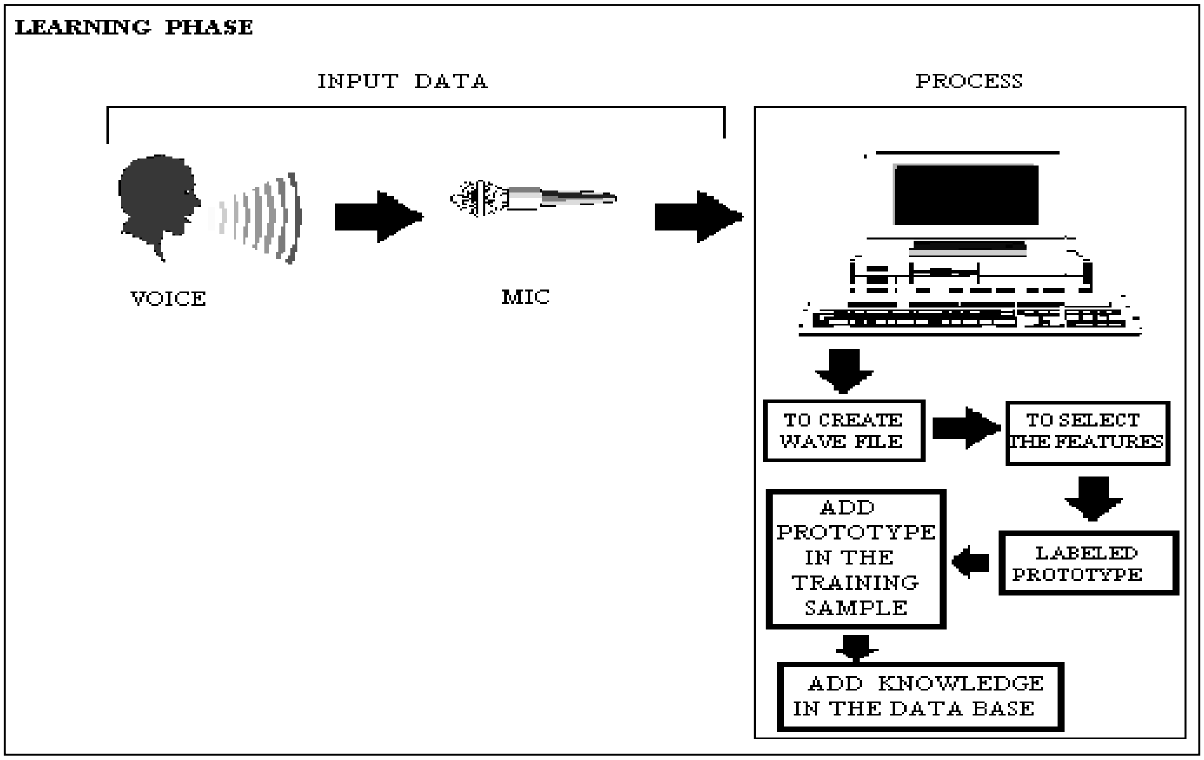

Figure 1. Learning Phase

3.2.- The Classification Phase for The Knowledge-Based System Model

The classification phase, was carried out in two following variants:

Variant 1: Using Cross Validation. In the dataset VOICES-ROUTINES, five-fold cross validation was employed (80% for the training sample and 20% for a test set). In the Table 3 the results obtained of the classification using the rules: 1-NN, k-NN, k-NCN and the neural network (BAP) are shown. It is observed in the Table 3, that the classification criterions are the geometric mean (*g*) and the global accuracy. The geometric mean has been proposed in some works [**16**, **17**] as evaluation criteria in the imbalanced training sample problem, when the training sample has only two classes. However, we propose a geometric mean for the evaluation from the imbalanced training sample problem, in the training samples that have more than two classes. We have considered the importance to know the accuracy by class in the training sample, which allows you to observe the geometric mean criteria. It is important to observe as the value of the *g* shows minor results as global accuracy, it happens because there are many errors in the minority class: Getto. However, when the *g* (and the accuracy) obtain major values, it happens because the minority class errors have been reduced. The best results with this Variant are shown in bold (Table 3), where it is observed that the new proposal Weighted k-NCN Classification has obtained the best performance.

Variant 2: In Real Time (Manually). This process is not automatic yet, thus it is necessary a user to initialize the beginning of the voice capture and to finalize this capture. The ideal system might be that when a sound or human voice it is listened, the capture begin automatically, and this capture will be finalized when there is not any sound. But, in this moment the automation is not included in our system. In Figure 2 the process used in the classification phase it is shown. This process consists of the identification of new prototypes that arrive to the system. A microphone captures these prototypes. They can only be captured one by one, the system does not allow capturing two or more prototypes at the same time. For the identification of a new prototype (frame), first it was captured the human voice by the microphone, immediately was saved in "wave" file extension. Once captured the human voice, this is converted in frames. These frames are obtained with the procedure above-mentioned (section 2.2). Then the wave file was opened to be applied the Frames Selection Procedure. Once labeled the frame by the classification rule, it is used the inference engine of the internal expert system to find in the knowledge base the text words corresponding to the new frame and to be able to do the language discrimination: Spanish or English. Once identified the language spoken, the classification rule (1-NN, k-NN or k-NCN according to be the case) is used to assign the label to the new prototype. Since the new prototype has class identification (label) already it can execute the mobile robot simulator with the routine concerning to the assigned class. For the identification of the other new prototypes and the execution of Ayllu´s routines, it was necessary to repeat this process.

In the experiment to 200 different human voices (woman and man) were captured, which have been classified using the rules: 1-NN, k-NN, k-NCN and neural network (BAP). These 200 voices are for each class, then globally to 1000 different voices were captured. In the Table 4 the results obtained in the classification phase with this Variant 2 it’s shown. The best results were obtained when we used the new proposal: Weighted k-NCN Classification.

TABLE 3. CLASSIFICATION PHASE WITH A VARIANT 1. IN THIS TABLE THE ACCURACY IN TWO CRITERIONS ARE SHOWN. ALSO, THE STANDARD DESVIATION **S[.]** FOR EACH CRITERION ARE SHOWN. FIGURE IN PARENTHESIS THE k VALUE USED.

CLASSIFIER	VOICES-ROUTINES			
	(g)	Accuracy	**S[(g)]**	**S[**Accuracy**]**
k-NN				
Euclidean Distance Function	64.67 (11)	87.28 (11)	0.2111	10.6858
Weighted Distance Function	66.06 (7)	89.81 (7)	0.1850	9.6750
k-NCN				
Euclidean Distance Function	66.90 (3)	89.17 (3)	0.2314	12.1315
Weighted Distance Function	**69.04** (3)	**92.05** (3)	*0.1838*	*9.6657*
Multi-Layer Perceptron BackPropagation	(g)	Accuracy	**S[(g)]**	**S[**Accuracy**]**
One Hidden-Layer (21 nodes)	54.65	58.82	0.3215	10.8567
Two Hidden-Layer (21 nodes)	54.30	58.05	0.2510	10.8548
Three Hidden-Layer (21 nodes)	54.60	58.70	0.2720	11.8672
One Hidden-Layer (100 nodes)	50.20	56.40	0.3208	13.1512
Two Hidden-Layer (100 nodes)	49.80	57.92	0.4210	10.6981

TABLE 4. CLASSIFICATION PHASE WITH THE VARIANT 2.

CLASSIFIER	VOICES-ROUTINES												
	# Errors by Class					# Well-Classified by Class					Global Errors	Global Well-Classified	# New Prototypes (by Class)
	C0	C1	C2	C3	C4	C0	C1	C2	C3	C4			
k-NN (k=3) Euclidean Distance Function	120	140	90	140	50	80	60	110	60	150	540	460	200
Weighted Distance Function	120	140	100	130	30	80	60	100	70	170	520	480	200
k-NCN (k=3) Euclidean Distance Function	110	140	90	120	30	90	60	110	80	170	490	510	200
Weighted Distance Function	0	130	80	90	10	200	70	120	110	190	**310**	**690**	200
Multi-Layer Perceptron BackPropagation (One Hidden-Layer and 21 nodes)	100	150	110	190	70	100	50	90	10	130	620	380	200

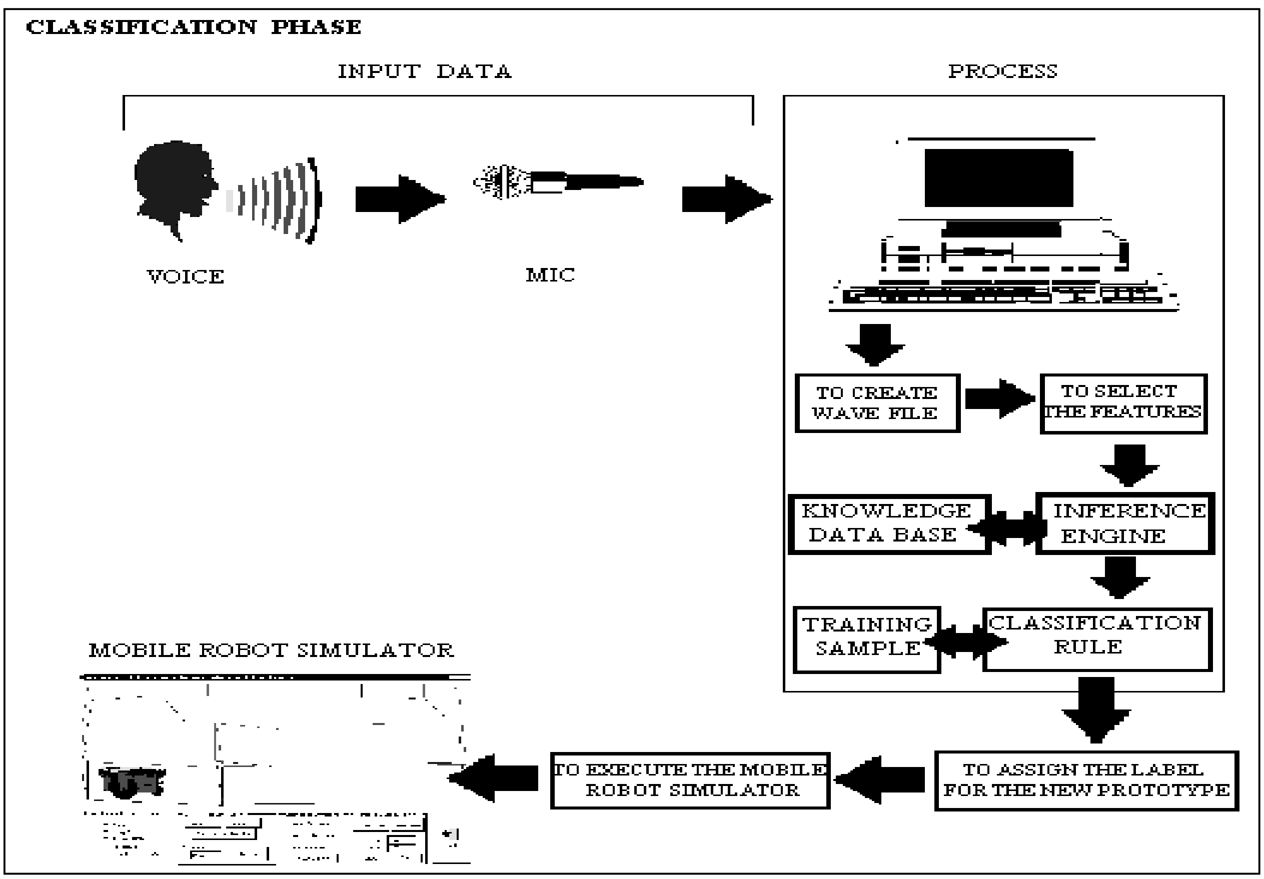

Figure 2. Classification Phase

4. Conclusions and Future Works

The new proposal Weighted k-NCN Classifier obtained the best results in most of the cases. However, the k-NN rule has obtained similar results in comparison with the new proposal. In addition, the neural network (BAP) has not shown to be very efficient for the imbalanced training sample problem, although the neural network (BAP) is very fast in the classification phase, in this work we observe that the execution times with the NN rules (k-NN and k-NCN) are as very similar as the neural network. Then, the execution times are very similar in all the techniques.

A future work is improving the classification phase in the Variant 2, applying some techniques such as: Editing [**22**, **23**], Depuration [**24**] and Reject Option [**25**]. Also, to realize an automation of the knowledge-based system allowing to detect the input data microphone without help from the user. However, it will be necessary changing the mobile robot simulator, by a real application with the robot: Pioneer DX2 of ActivMedia.

References

[**1**].-Scarabino, JC (2000): "Sistemas Expertos: Aspectos técnicos", [en línea] *5campus.org, Inteligencia Artificial* <http://www.5campus.org/leccion/sistexp at> [23/Mayo/2002].

[**2**].-Sangster A. and Wilson A. (1991). "*Knowledge Based Learning".* Expert Systems for Information Management, Vol. 4, No 1, pp. 49-63.

[**3**].-D. E. Rumelhart and J.L. McClelland (1987) "*Parallel Distributed Processing: Explorations in the Microstructure of Cognition*". Vol I, II and III. Cambridge. MA: MIT Press.

[**4**].-A.Benallegue and A.Bennani-Hassa. (1993) Adaptive controller for robot manipulators with elastic joints using passive feedback systems approach. In *Proc. of the IMACS/IFAC 2nd Int. Symposium on Mathematical and Intelligent Models in System Simulation*, pages 12-16.

[**5**].-Barandela, R. and M. Juarez (2001). "Ongoing Learning for Supervised Pattern Recognition". *Submitted to SIBGRAPI-2001, Brazil.*

[**6**].-De Lope Asiaín J. (1998). "*Mobile Robot Simulator*", [en línea]. Departamento de Sistemas Inteligentes Aplicados Universidad Politécnica de Madrid. <http://www.sia.eui. upm.es/~jdlope/mrs/worlds.shtml>[23/Mayo/2002].

[**7**].-ActivMedia Robotics (2002). "Intelligent Mobile Robots & Bases". [En Linea] ActivMedia Robotics, LLC. <http://www.activrobots.com/ROBOTS/ index.html> [23/Mayo/2002].

[**8**].-ActivMedia Robotics (2002). "ActivMedia Robotics Interface for Applications". [En Linea] ActivMedia Robotics, LLC. <http://www .activrobots. com/ SOFTWARE/ index. html> [23/Mayo/2002].

[**9**].-ActivMedia Robotics (2002). " Ayllu Downloads Distributed Behavior-Based Control Ayllu Manual v 1.5". [En Linea] ActivMedia Robotics, LLC. <http://activmedia.com/> [23/Mayo/2002].

[**10**].-Moreno L. J. (2000). "*Desarrollo de un reconocedor de dígitos con distinción de énfasis: Capítulo II. CSLU ToolKit - Reconocimiento de voz basado en Frames*". Universidad de las Américas-Puebla, Escuela de Ingeniería Departamento de Ingeniería en Sistemas Computacionales. <http:// mailweb. udlap.mx/~tesis/lopez_m_j/capitulo2.html> [23/Mayo/ 02].

[**11**].- Lleida S. E. (2000). "*Reconocimiento Automático del Habla*". Centro Politécnico Superior-Universidad de Zaragoza. <http:// www.gtc.cps.unizar.es/~eduardo/>[23/May/02]

[**12**].- Hosom, J.P., Cole R., and Fanty M. (1999). "*Speech Recognition Using Neural Networks at the Center for Spoken Language Understanding*". NSF Graduate Research Traineeships award (grant number 9354959) and the CSLU Member companies. <http://cslu.cse.ogi.edu/tutordemos/nnet_recog/recog.html>[23/Mayo/2002].

[**13**].- Cid, A. (1999). "*Uso de las RNA para el reconocimiento de voz*". International Sensible Artificial Intelligence Network of Thoughts. <http://www.intersaint.org/acid/rvpm5. htm> [23/Mayo/2002].

[**14**].- Carreira-Perpinan, M. A. (1995): Compression neural networks for feature extraction: Application to human recognition from ear images (in Spanish). MSc thesis, Faculty of Informatics, Technical University of Madrid, Spain.

[**15**].- J. Tani (1996)."Model-based Learning for Mobile Robot Navigation from the Dynamical Systems Perspective," *IEEE Trans. on System, Man and Cybernetics Part B (Special Issue on Robot Learning)*, Vol.26, No.3, pp. 421-436.

[**16**].- Kubat, M. and S. Matwin. (1997). "Addressing the Curse of Imbalanced Training Sets: One-Sided Selection". *Proceedings of the 14th International Conference on Machine Learning,* 179-186.

[**17**].- Barandela, R., Sánchez J. S., García, V., Rangel, E., (2001). "Fusion of techniques for handling the imbalanced training sample problem". In: *Procedings of 6th Ibero- American Symposium on Pattern Recognition*, Brasil, 31-40.

[**18**].- Bruzzone, L. and Serpico, S.B.,(1997). "*Classification of Imbalanced remote- sensing data by neural networks*". Elsevier Science B.V., 0167-8655, 97. PH S0167-8655 (97) 00109-8.

[**19**].- T. Kohonen (1995). Self-Organizing Maps, Springer-Verlag: Berlin, Germany.

[**20**].- Chaudhuri, B. B. (1996). "A New definition of neighborhood of a point in multi-dimensional space". *Pattern Recognition Letters* 17, 11-17.

[**21**].- J.S. Sánchez, F. Pla and F.J. Ferri (1997). On the use of neighborhoods-based non-parametric classifier. *Pattern Recognition Letters,* Vol. 18, pp. 1179-1186.

[**22**].- Wilson , D. L. (1972). "Asymptotic Properties of Nearest Neighbor Rules using Edited Data", IEEE Transactions on Systems, Man and Cybernetics **2**(3) , 408-421.

[**23**].- Sánchez, J. S., Barandela, R., Marques, A.I., Alejo, R., Badenas, J. (2001). "Análysis of new techniques to obtain quiality training sets". *Preprint submitted to Elsevier Science* 29 October.

[**24**].- Barandela, R., and E. Gasca (2000). "Decontamination of training samples for supervised pattern recognition methods". *In: Advances in Pattern recognition, Lecture Notes in Computer Science, vol. 1876, F. Ferri et al. (eds.), Springer,* Berlin*, 621-630.*

[**25**].- Barandela, R. and T. Najera (2002). "Supervised Pattern Recognition with Incomplete Training Samples". Enviado a: Workshop on Statistical Pattern Recognition, Canadá.