AIBO Talking Procedure based on Incremental Learning Approach

Zlatko Fedor, Peter Sinčák

**Center for Intelligent Technologies,
Department of Cybernetics and Artificial
Intelligence, FEI TUKE Kosice
Home page: http://www.ai-cit.sk**

Abstract. The target of this project is to propose and implement incremental system for linguistic command recognition in multi agent system MASS, based on client-server architecture. Preprocessing is realized with the aid of Mel-frequency cepstral coefficients and classification is realized by modified MF Artmap. System allows remote parallel learning of various commands, their consecutive identification and robot dog AIBO control.

Keywords: recognition, words, MF Artmap, MFCC, multi-agent, client-server, incremental, system, MASS, AIBO

1. Project Definition and Task Determination

The goal of this project is to propose and implement incremental system for linguistic command recognition in multi-agent system MASS with Adaptative Resonance Theory (ART) like methods especially with modified MF Artmap and sound preprocessing which is realized with the aid of Mel-frequency cepstral coefficients ([1], 2006). Finally, the chosen methods in form of plugins are tested with this system.

2. The State of the Art in the Domain

If we want to solve some problems in real life with methods of artificial intelligence, we use very often recognition and classification. These concepts are very similar but there are slight differences between them. While in process of classification the number of classification classes is known, in recognition process these classes are being created during the recognition process. The concept of classification can be defined as follows: Incorporation of objects or events into specific classes by the decision rule. The objects which are familiar enough are incorporated into the same class. Generally the classification rule has some parameters which are changeable. This change of

parameters is the training of the classification tool. In the domain of linguistic command recognition, the neural networks are the common classification tools and the recurrent types of them with adaptive resonance are the best choice in many cases.

3. Selected Methods and Approaches

Modified MF Artmap is derived from the existing MF Artmap ([8], 2002) model. It utilizes all advantages of original MF Artmap, for example speed of learning/classification and identification of unknown classes. Moreover some errors from this neural network have been removed.

First modification was the change of work with parameter R on comparative layer network. In the original network it performs check of distance from the central cluster for every dimension separately. Original network is using the same parameter R for all dimensions and clusters.

It brings the following disadvantages:

1. Clusters have very similar measurements and they can't have different size in different dimension.
2. Creation of extra clusters which are not needed.
3. The occurrence of unclassified inputs even if they belong to certain clusters.

These problems were solved as follows. Comparison distance from the centre cluster for every dimension is done with use of parameter R which is different for every dimension and cluster. Then with the creation of a new cluster, R parameters are assigned for every dimension. The parameters are from interval <0, 1>. Second modification is the change of update logic for parameter R. At first parameter q is updated by formula:

$$qn = qs + 1$$

where qs is the count of examples in cluster before addition of new example, qn is the count of examples in the updated cluster.

Next step is to update parameter X for every dimension. Update of this parameter is by original MF Artmap formula:

$$Xn = Xs + \frac{1}{qn} \cdot (Xs - X)$$

where Xn is the position of the new centre cluster in actual dimension, Xs is the value original cluster centre, X is position of the new example, qn is the count of examples in cluster.

Finally for every dimension of cluster parameter R is updated with of this formula:

$$Rn = Rs + \frac{sign \cdot \left| |Xn - X| - |Xs - X| \right|}{qn}$$

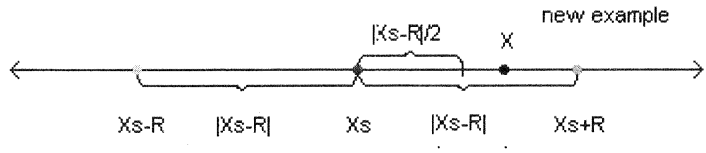

Fig. 1. Update of the dimension cluster

where Rn is new radius of the cluster for actual dimension, R is old cluster radius, qn is the count of examples in actual cluster, |Xn- X| is distance of sample from the centre cluster, |Xs- X| is distance of sample from the cluster center before change, parameter sign is described here by the formula:

$$sign = \begin{cases} -1, \frac{|Xs - R|}{2} > |Xs - X| \\ 1, otherwise \end{cases}$$

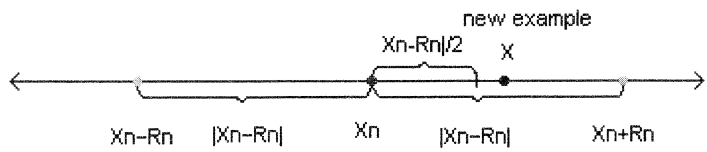

Fig. 2. Updating process for the modification of the cluster parameters

Next modification of the network was done to solve error situation in original network which occurs when new sample falls in the middle of some cluster but had another class. In this case, that cluster is deleted from the network.

4. Design and Implementation

Everything from the previous part was implemented as a plugin for MASS. This part will describe MASS and plugin types which are used in this system.

MASS could be described as multi-agent, incremental, plugin system with client-server architecture. With plugins for object recognition it is possible to learn various objects or to recognize them in parallel manner for many clients around the world. Gained knowledge will be stored on server which will host the object recognition setup in MASS.

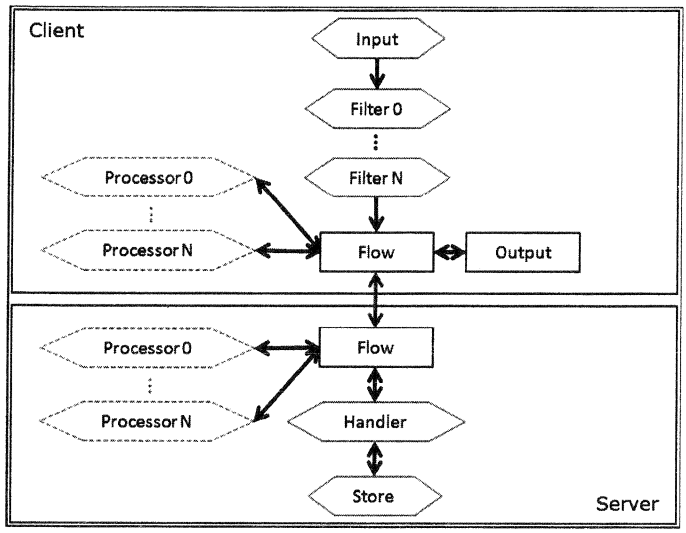

Fig. 3. MASS Plugin system

Fig. 1 describes the plugin system of MASS. Red plugin types are required and together they represent the smallest possible system. Green plugin types are optional. As you can see there can be more filter and processor plugins in the system and processor plugins can by placed whether on client or server side.

Flow plugin is special because the same plugin is placed on both sides and these sides are communicating together. Each plugin type will be briefly described in the sequel.

Flow is required part of the system. This plugin type is managing the client-server communication. If some user input is needed, form is included in the plugin.

Output is required part of the system. Its task is to provide and react on results provided by server. The reaction could be manipulation with some connected robot or device.

Input is optional part of the system. It provides input data from devices like webcams, microphones and sensors.

Filter is optional part of the system. It modifies its input, which can be from Input or other Filter plugin. The purpose of its use is similar to that of Filters known from image or audio processing.

Processor is optional part of the system. It can be placed whether on client or server side of the system. Processor should change the input data to data which can be used in classification, clustering or other types of Handler plugin tasks.

Handler is optional part of the system. This plugin should have all the functionality required for manipulation with data in database. Classification, clustering and other similar operations should be implemented in this type of plugin.

Store is optional part of the system. Here should be implemented everything related to data storage. This could be implemented all by authors or they can implement link to SQL or similar database system. Moreover also internet can be considered as some sort of database.

5. Experiments

This section presents experiments on robotic dog AIBO in Slovak language. Experiments are using aibo in "remote" regime, when robot could be controlled over wifi interface. Plugin InputOutputAudioAibo allows to read the audio data from the robot microphone and to send him commands that are dog performs.

5.1. Training

Experiments are using following verbal commands from four speakers. Commands were spoken by three men and one woman. Total word count for training was 165. Individual verbal commands were spoken directly to robot AIBO from approximately one meter distance without disturbing environment sound. In the next table is the count of recorded commands from individual speakers:

commands in Slovak language	count				
	speaker 1 (man)	speaker 2 (man)	speaker 3 (woman)	speaker 4 (man)	total count
sadni	5	4	5	3	17
ľahni	5	2	5	4	16
vstaň	4	3	3	5	17
tancuj	5	3	3	5	16
kopni	5	3	4	2	14
doprava	6	3	5	5	19
doľava	5	4	3	4	16
dopredu	5	3	4	4	16
dozadu	5	5	4	5	19
lez	5	3	3	4	15

5.2. Testing

The test set consists of 58 verbal commands. Speakers were the same from the training stage. Commands were spoken directly to aibo at approximately one meter distance without disturbing environment sound. Count of the commands are showed in the next table:

commands in Slovak language	count				
	speaker 1 (man)	speaker 2 (man)	speaker 3 (woman)	speaker 4 (man)	total count
sadni	1	1	0	3	5
ľahni	2	2	0	2	6
vstaň	2	1	2	2	7
tancuj	1	1	2	1	5
kopni	2	1	1	3	7
doprava	2	1	1	1	5
doľava	2	1	2	1	6
dopredu	1	1	1	2	5
dozadu	2	0	1	1	4
lez	3	1	2	2	8

5.3. Results

Parameter R for the neural network was 0.6. Next table shows the classification percentage of testing commands.

actual class	predicting class									
	Sadni	ľahni	vstaň	tancuj	kopni	doprava	doľava	dopredu	dozadu	lez
sadni	100	0	0	0	0	0	0	0	0	0
ľahni	16.7	83.3	0	0	0	0	0	0	0	0
vstaň	0	0	100	0	0	0	0	0	0	0
tancuj	0	0	0	100	0	0	0	0	0	0
kopni	0	0	0	14.3	85.7	0	0	0	0	0
doprava	0	0	0	0	0	60	40	0	0	0
doľava	0	0	0	0	0	33.3	66.7	0	0	0
dopredu	0	20	0	0	0	0	0	80	0	0
dozadu	0	0	0	0	0	0	0	0	100	0
lez	0	0	0	0	0	0	0	0	0	100

Final classification accuracy is 87.93%.

In the third experiment the cycle count was increased to 5. Results are in the next table.

actual class	predicting class										
	sadni	ľahni	vstaň	tancuj	kopni	doprava	doľava	dopredu	dozadu	lez	unknown class
sadni	100	0	0	0	0	0	0	0	0	0	0
ľahni	16.6	50	0	16.6	0	0	16.6	0	0	0	0
vstaň	0	0	100	0	0	0	0	0	0	0	0
tancuj	0	0	0	100	0	0	0	0	0	0	0
kopni	0	0	0	0	100	0	0	0	0	0	0
doprava	0	0	0	0	0	80	0	0	0	0	20
doľava	0	0	0	0	0	16.6	83.4	0	0	0	0
dopredu	0	0	0	0	0	0	0	80	0	0	0
dozadu	0	0	0	0	0	0	0	25	75	0	0
lez	0	0	0	0	0	0	0	0	0	100	0

Final classification accuracy was been 86.2%.

6. Contribution to the Results in the Domain

From the experiments you can see that similar words in way of pronunciation could confuse the network. One confusing example are words "doľava" and "doprava" which caused bad classification quite often. This special case could be solved by teaching only "ľava" and "prava". This also shows the robustness of given system.

7. Conclusion

This work is using modified version of MF Artmap neural network which reach better results in comparison with original MF Artmap network. It was showed in experiments with almost 88% of classification accuracy. System MASS allowed simple implementation of individual methods that were needed for the recognition in form of plugins.

Output of this work is modified MF Artmap network, plugins for recognition of isolated words for system MASS. After system startup of MASS, people for all over the world can teach system new words and improve the quality of recognition.

Next research could improve recognition with additional improvements of the neuronal network. Moreover, it could be convenient to implement that system will ask for class of the word which is not learned and it will learn it afterwards. It would be good to create filter that will be able to remove disturbing environment sounds and focus only on speaker voice.

Acknowledgement:

Research supported by the National Research and Development Project Grant 1/0885/08 "Learnable Systems based on Computational Intelligence" 2008-2010

References

[1] Psutka, J.: *Mluvíme s počítačem česky*. Academia, Praha (2006)

[2] Nuttall, A. H.: Some *Windows with Very Good Sidelobe Behavior*, IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol.ASSP-29, No.1, February, (1981). Dostupné na internete: http://en.wikipedia.org/wiki/Window_function#Hamming_window

[3] Černocký, J., Burget L: *Parametrizace reči*. FIT VUT Brno. Dostupné na internete: http://www.fit.vutbr.cz/~cernocky/speech/pred/11_priznaky/11_priznaky.pdf

[4] Olajec, J.: *Návrh rozpoznávania izolovaných čísloviek.*, 2004. Department of Telecommunications, EF ZU Zilina. Dostupné na internete: http://kt.uniza.sk/~olajec/publications/2004_Jan_Olajec_Diplomova_praca.pdf

[5] Psutka, J., Muller, L.: *Optimization of Same Parameters in the Speech Parameters in the Speech-Processing Module Developed for the Speaker Independent ASR System*. In: Proc. Of IIIS 2003, Orlando, USA, (2003), s. 414-418

[6] Sinčák, P., Andrejková, G.: *Neurónové siete*. Inžinierske aplikácie II. Dostupné na internete: http://www2.fiit.stuba.sk/~cernans/nn/nn_download/Sincak_Andrejkova_vol_2.pdf

[7] Olajec, J., Jarina, R.: *Použitie metódy 3TDCM pri rozpoznávaní izolovaných slov neurónovou sieťou*, IEEE Vršov 2005, October 2005, Vršov, Cech Republic, ISBN 80-214-3008-7. Dostupné na internete: http://kt.uniza.sk/~olajec/publications/%5B02%5D_-_Vrsov_2005.pdf

[8] Hric M.: *Integrácia neurónových sietí typu ARTMAP s prvkami fuzzy systémov pre klasifikačné úlohy*. Dostupné na internete: http://www.ai-cit.sk/source/publications/thesis/master_thesis/2000/hric/html/index.html

[9] Pai, H. F., Wang H. C.: *A study of the two-dimensional cepstrum approach for speech recognition*, Computer Speech and Language vol.6 (1992)

[10] Ariki, Y., Mizuta, S., Nagata, M., Sakai, T.: *Spoken-word recognition using dynamic features analysed by two-dimensional cepstrum*, IEE Proceedings, vol.136 (1989)

[11] Hudec, M.: *Prehľad problematiky*. Dostupné na internete: http://www.elajnus.sk/diplomovka/files/prehlad.pdf

[12] Biswas, S., Ahmad, S., Islam Mollat, M.K.: *Speaker Identification Using Cepstral Based Features and Discrete Hidden Markov Model*, Information and Communication Technology, (2007). ICICT apos;07

[13] Molnárová, M., Spalek, J.: *Fuzzy monitoring of the safety-related critical processes*, In: Híradástechnika 9/2001, Vol. LVI., ISSN 0018-2028, Budapest, pp. 21-24

[14] Molnárová, M., Spalek, J., Šurín, P.: *The Use of Fuzzy Logic for Safety-Related Decision*, In: IFAC Conference Control Systems Design, Bratislava, 18.-20. June (2000), pp. 548-553

[15] Hájek, P, Olej, V. : *Municipal Creditworthiness Modelling by NEURAL Networks*, Accepted to Acta Electrotechnika , Informatika TU Košice