

Hybrid Rule and Neural Network based Framework for Ubiquitous Computing

Andrey V. Gavrilov

*Department of Production Automation in Machine Building,
Novosibirsk State Technical University*

Karl Marx Av., 20, Novosibirsk 92, 630092 Russia

Andr_gavrilov@yahoo.com

Abstract

One of most perspective techniques for sensing in ubiquitous computing systems is neural networks. But for a prior knowledge representation it is most appropriate to employ rule based techniques. So usage of hybrid intelligent systems based on combination of neural networks and rule based techniques seems perspective for development of smart environment. In this paper features of hybridization in smart environment are described and usage of technology of hybrid expert systems based on shell ESWin is proposed.

1. Introduction

Ubiquitous computing [1], [2], [3], [4] (just as intelligent robotics) is a field of AI in which all problems of AI are concentrated and not only AI.

In order to realize ubiquitous pervasive computing environment, three technology areas are required:

- 1) sensing technology where information on user and surrounding environment are perceived and collected, in particular location sensing technology [5], [6],
- 2) context aware computing [7], [8], [9] technology where such information are processed and properly presented to users as different services,
- 3) network technologies where information are collected from sensors and distributed to customers – services and users [10].

It is possible pick out following main features of ubiquitous computing systems (UCS):

- 1) distribution of obtaining and processing of sensor information,
- 2) variety of information needed processing,
- 3) necessity of learning during interaction with environment, in particular, in respect to

existing of unexpected events and objects needed for including into processing,

- 4) essential role of different kinds of human-machine interaction,
- 5) high requirements to security,
- 6) data processing in real time,
- 7) wide usage of embedded processing units.

One of most perspective technologies for sensing is neural networks [11], [12], [13].

There are following tasks for neural networks in development of ubiquitous computing systems:

- 1) perception, i.e. recognition of objects and changes in environment, in particular, invariant recognition of moving objects, e.g. recognition of gesture, position and emotions of human beings,
- 2) clustering and recognition of events and scenarios (sequence of events in time),
- 3) prediction of future events and situations,
- 4) indoor localization of mobile devices.

The following tasks are more appropriate for implementation in rule based techniques:

- 1) a prior description of behavior of smart object with respect to perceived objects/situations and context,
- 2) reflex to perceived important situation starting determined behavior,
- 3) diagnostics of sensor network.

These two different paradigms (rule based and neural network based systems) may be combined in hybrid approach to development of intelligent systems [14], [15].

In this paper we propose for development of hybrid intelligent systems of smart environment usage of paradigms implemented in expert shell ESWin [16], [17] combining with neural networks. Previously this shell was oriented to development of dialog expert systems for solving of tasks for diagnostics and

recommendations in well structured areas. But now this one is improving and expending for building of embedded systems in smart environment.

Proposed architecture of hybrid intelligent system may be named as both stand-alone and loosely-coupled model with accordance to proposed in [18] classification.

2. Knowledge representation and processing

Knowledge representation in ESWin (fig. 1, fig. 3) [16], [17] is based on frames, fuzzy rules and linguistic variables [19]. Frames are used for description of structure of world (area) including properties and hierarchies of entities (frames-classes). Besides frames aim to store current facts about entities (frames-instances). Rules are used for description of solving of tasks, e.g. behavior of any smart object. Linguistic variables are used for description of fuzzy properties which may have both quantitative and qualitative values at the same time.

Solving of task is a fuzzy rule based backward chaining using data from frames. Backward chaining provides execution of any behavior or decision making depending on current set of facts and goal. Goal means what we want to get from this behavior. For example, it may be name of behavior.

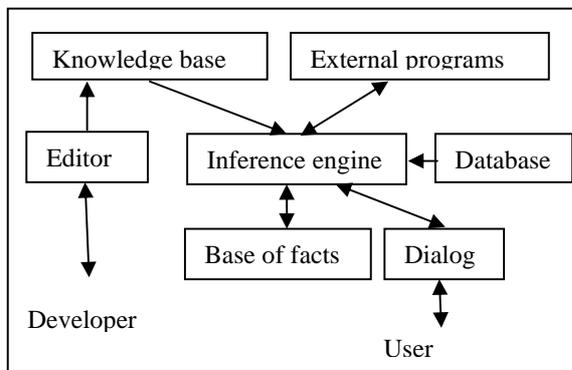


Figure 1. Architecture of ESWin

Let take up ubiquitous service helping to find and operate the objects in room for elderly or handicapped people. Assume that we have one or more camera for monitoring of location and state of person. Below you see fragment of knowledge base of this service realized in knowledge representation language of ESWin, modified for smart environment.

```
// This frame describes possible goals for backward
// inference
```

```

Frame = Goal
    Find: (Cornflakes box; Book; TV control;
          Medicine; thermometer)
EndF
// These frames are in knowledge base (model of
// world)
Frame = Cornflakes box
    Parent: Cereal box
EndF
Frame = Cereal box
    Parent: Box
    Action: (pour out; fill)
EndF
Frame = Box
    Parent: Object
    Fullness: (full; empty)
    Openness: (opened; closed)
    Action: (open; close)
EndF
Frame = Object
    Name:
    Action: (take; put; find)
    Location: (into; visible; anywhere)
    Into:
    // Here "lv" is a slot's type "linguistic variable"
    Where (lv): (near; far; not far; close by)
    Direction: (ahead; to left; to right; to back)
    Path:
    Color: (Red; Green; Yellow; Blue; White;
            Black)
    Size (lv): (huge; large; small; tiny)
    Form: (Sphere; Cube; Rectangle; Complex)
EndF
// This and above frame provides generation of frame-
// fact // from slot Name of Frame-parent with Name like
// value
// of corresponding slot of Object with inheritance of
// properties of Object
Frame = Name
    Parent: Object
EndF
Frame = Person
    Action: (turn to right; turn to left;
            forward; stop; turn back; go to; find)
    State: (stand; lie; sit; walk)
    Direction of view:
EndF
// These frames are in base of facts
Frame (instance) = Cornflakes box
    Color: Green
    Location: into
    Into: Cupboard
EndF
// Facts about visible (recognized) objects
  
```

```

Frame (instance) = Person
    State: stay
EndF
// Distances to visible objects
Frame (instance) = Cupboard
    Where: 1, 7
    Direction: ahead
EndF
Frame (instance) = Window
    Where: not far
    Direction: to left
EndF
// Rules for prompting to find any object being into
// cupboard
Rule 1
    =(Object.Name; *Object)
    =(Object.location; into)
    =(Object.into; Object_2)
    =(Object_2.path; any)
    =(Object_2.direction; ahead)
    <(*Object_2.Distance; 0,3)
DO
    =(Goal.Find; *Object)
    MS(Action.Ms; Stop, please take *Object
from
    *Object_2)
EndR
Rule 2
    =(Object.into; Object_2)
    =(Object_2.Direction; ahead)
    <>(*Object_2.Distance; close by)
DO
    =(Object_2.Path; ahead)
    MS(Action.Ms; Go straight ahead to
cupboard)
EndR

```

Note that identifier **Object* is analog of identifier of variable which replaced by corresponding value from previous steps of interpretation of rule.

In this example some data (values of frame's slots), such as *Direction* and *Distance* to any objects-places (*Cupboard* or *Window*), *Form* of *Object* (*Sphere*, *Cube*, and so on), *Color* (*Green*, *Red* and so on), *State* and so on, may be obtained from sensing subsystem based on one or more neural networks. Any of them may be produced by rules, for example, *Cupboard.Direction* may be got from *Person.Direction of view* by executing of rule:

```

Rule 3
    =(Person.Direction of view; *Object)
DO
    =(Object.Direction; ahead)

```

To get more appropriate language for smart environment the knowledge representation language of ESWin must be modified as follow:

1) To provide more compact knowledge base with description of different behaviors it is needed to introduce concept of variables-references in rules which may assume value (replaced by value similar to macros) during interpretation of rules until task is solved (e.g., reference **Object* and **Object_2* in our example).

2) To provide capability of solving of task during long time it is needed to introduce two modes of interpretation (inference) – short time (for decision making in one moment) and long time (decision making during monitoring and control like in our example).

3) To provide capability to control of several processes simultaneously (e.g. our fragment of knowledge and any reaction for recognized important situation) it is needed to introduce into interpreter multi-task capability. It means that start of any task (chaining) causes start of process. And we have to support interaction between processes, in particular, interruption, waiting and deletion of processes. In above described example the interpretation of Rule 1 may be stop and wait until condition $\langle \text{Cupboard.Distance; } 0,3 \rangle$ or “distance between person and cupboard $< 0,3$ meters” is true. It means that process will wait when person would come to cupboard. It is obvious that we have to implement any mechanism for interruption and deletion of this process.

4) To provide reactive capabilities it is needed to implement forward chaining (in ESWin it is absent). Although it is possible to provide reactive capability by backward chaining with multi-task capability and waiting of respect condition this simulation of reactive response is not suitable and effective because we need to start chaining for every possible situation and corresponding response.

3. Interaction between inference and neural networks

To support of interaction between rule-frame-based inference and neural networks the following mechanism was designed.

Special data structure SOURCE describes necessary parameters for the determination of interaction of expert shell with other program, in particular, neural network based sensing routine. If knowledge base interpreter will find a special frame in the condition of checking rule in the process of

inference, it realizes a stop of process and sends an inquiry for the reception of fact from other program. After receipt of fact a process of inference continues. If inquired fact does not enter from the external source in specifying gap of time, the fact is taken as empty and the inference continues. But if this special frame is used in conclusion rules, the expert system sends a fact to other program. The example of this structure for connection with neural network by socket is

SOURCE = NN1

Type: Connection

Port: 1234

IP: 127.0.0.1

TimeOut: (10) // time of waiting of response

Old: (60) // time for aging of fact got from NN1

EndF

This structure contains all information needed for interface with neural network with name NN1 by socket. Usage of socket allows deploying modules of this hybrid intelligent system in network of smart environment.

Variables which values produced by neural network are described in frame with same name like this structure:

FRAME = NN1

Form: (Cube; Sphere; Rectangle; Complex)

Color: (Green; Red; Yellow; Blue; White; Black)

ENDF

If we use forward inference then the neural network is able to produce fact in arbitrary moment of time when it recognizes any situation or object. In both cases of inference (backward or forward) the result of execution of neural network is fact understandable for inference engine.

A dictionary for interface between neural network and inference engine is used. It is employed for coding of words (phrases) contained in frame for interface as vectors for neural network and for decoding of vectors obtained from neural network as words (phrases) constructing facts. In our example (described above) we must have in dictionary following words: form, cube, sphere, complex (using for recognition of form), state, stay, lie, sit (for recognition of state of person), object, cupboard, cornflakes box, cup, bottle, window (for recognition of objects).

Note that if any fact checked by rule is already in base of fact (got from neural network earlier) then this fact is used without execution of neural network while time from one's appearance less than *Old*.

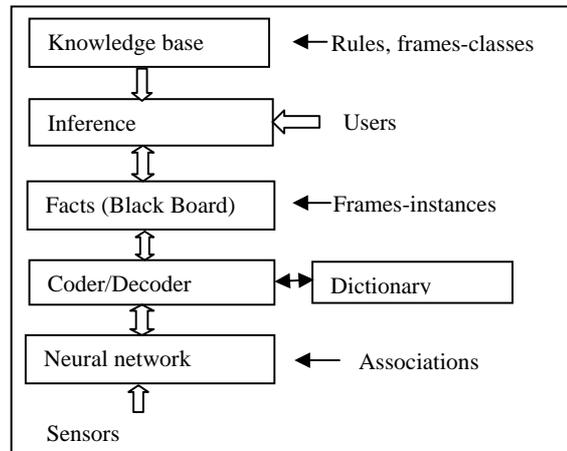


Figure 2. Layers of information processing

4. Conclusions

In this paper a framework for implementation of AI in smart environment is proposed based on hybridization of rule based systems for decision making and monitoring and neural networks for sensing. Example of knowledge representation in such AI based on modified language of expert shell ESWin ([16], [17]) is shown. Features of required modification of usual expert shell, in particular ESWin are described. Interaction between ESWin and neural networks is implemented by sockets. Thus it is possible to implement distributed intelligent system.

Now new version of ESWin with modifications for usage in smart environment is developing.

5. References

- [1] Ju-Jang Lee, Kap-Ho Seo, Changmok Oh and Z. Zenn Bien, "Development of a future Intelligent Sweet Home for the disabled", *Artificial Life and Robotics*, 11, 2007, pp. 8-12.
- [2] M. Weiser, "The Computer for the 21st Century", *Scientific America*, Sept. 1991, pp. 94-104; reprinted in *IEEE Pervasive Computing*, Jan.-Mar., 2002, pp. 19-25.
- [3] M. Satyanarayanan, "Pervasive Computing: Vision and Challenges", *IEEE Personal Communications*, Aug. 2001, pp. 10-17.
- [4] Jianhua Ma et al, "Towards a Smart World and Ubiquitous Intelligence: a Walkthrough from Smart Things to Smart Hyperspaces and UbiKids", *J. Pervasive Computing & Comm.*, 1(1), March, 2005, pp. 53-68.
- [5] G.Timar et al, "Sensing-Computing-Actuation in a Multi-Target Tracking Framework", *Proc. of int. conf. ECCTD'03*, vol. II, Crakow 2003, pp. 77-80.
- [6] J. Hightower, G. Borriello, "Location Systems for Ubiquitous Computing", *IEEE Computer*, Vol. 34, No. 8., 2001, pp. 57-66.
- [7] Hung Q. Ngo, Anjum Shehzad, Saad Liaquat Kiani, Maria Riaz, Kim Anh Ngoc, and Sungyoung Lee, "Developing Context-Aware Ubiquitous Computing Systems with a Unified

- Middleware Framework”, *The 2004 International Conference on Embedded & Ubiquitous Computing (EUC2004)*, Springer-Verlag Lecture Notes in Computer Science, August 26-28, 2004, pp. 672-681.
- [8] N.Q.Hung, L.X.Hung, and Sungyoung Lee, "A Middleware Framework for Context Acquisition in Ubiquitous Computing Systems", *Second International Conference on Computer Applications (ICCA 2004)*, Myanmar, 8th January, 2004.
- [9] Saad Liaquat Kiani, Maria Riaz, Yonil Zhung, Sungyoung Lee, and Young-Koo Lee, "A Distributed Middleware Solution for Context Awareness in Ubiquitous Systems", *Proc. of 11th IEEE Int. Conf. on Embedded and Real-time Computing Systems and Applications (RTCSA 2005)*, 17-19 August 05, HongKong.
- [10] Ian F. Akyildiz et al. "A survey on Sensor Networks", *IEEE Communications Magazine*, August, 2002, pp. 102-114.
- [11] Uzair Ahmad, Andrey Gavrilov and Sungyoung Lee, "Modular Multi Layer Perceptron for Location Awareness", in *Proc. of 2006 IEEE World Congress on Computational Intelligence, Int. Joint Conf. on Neural Networks (IJCNN 2006)*, July, 16-21, 2006, Vancouver, BC, Canada, pp. 6497-6503.
- [12] Uzair Ahmad, Andrey V.Gavrilov, Sungyoung Lee and Young-Koo Lee, "Self-scalable Fuzzy ArtMap for Received Signal Strength Based Location Systems", *Soft Computing*, vol.12, N.7, May, 2008, pp. 699-713.
- [13] Andrey Gavrilov and Sungyoung Lee, "An Approach for Invariant Clustering and Recognition in Dynamic Environment", in *Proc. of IEEE Int. Conf. CISSE-2006*, in book: *Advances and Innovations in Systems, Computing Science and Software Engineering* (Ed. Khalet Elleithy), Springer, 2007, pp. 47-52.
- [14] Srefan Wermter and Ron Sun, *Hybrid Neural Systems*, Springer, Heidelberg, New-York, 2000.
- [15] Zili Zhang and Chegqi Zhang, "Agent-based Hybrid Intelligent Systems", *LNAI 2938*, Springer-Verlag, Berlin, Heidelberg, 2004.
- [16] A.V. Gavrilov and J.V. Novickaja "The Toolkit for development of Hybrid Expert Systems," *Proc. of 5-th Int. Symp. "KORUS-2001"*, Tomsk, TPU, 2001, vol. 1, pp. 73-75.
- [17] A.V. Gavrilov and N.A. Chistyakov, "An architecture of the toolkit for development of Hybrid Expert Systems", *Proc. of Int. Conf. IASTED ACIT-2005*, Novosibirsk, June, 2005.
- [18] L. R. Medsker and D. L. Bailey, "Models and Guidelines for Integrating Expert Systems and Neural Networks", in: A. Kandel and G. Langholz (Eds.), *Hybrid Architectures for Intelligent Systems*, CRC Press, Boca Raton, 1992, 154-171.
- [19] L.A. Zadeh, *The Concept of a linguistic variable and its application to approximate reasoning*, Elsevier Pub. Co., N.Y., 1973.

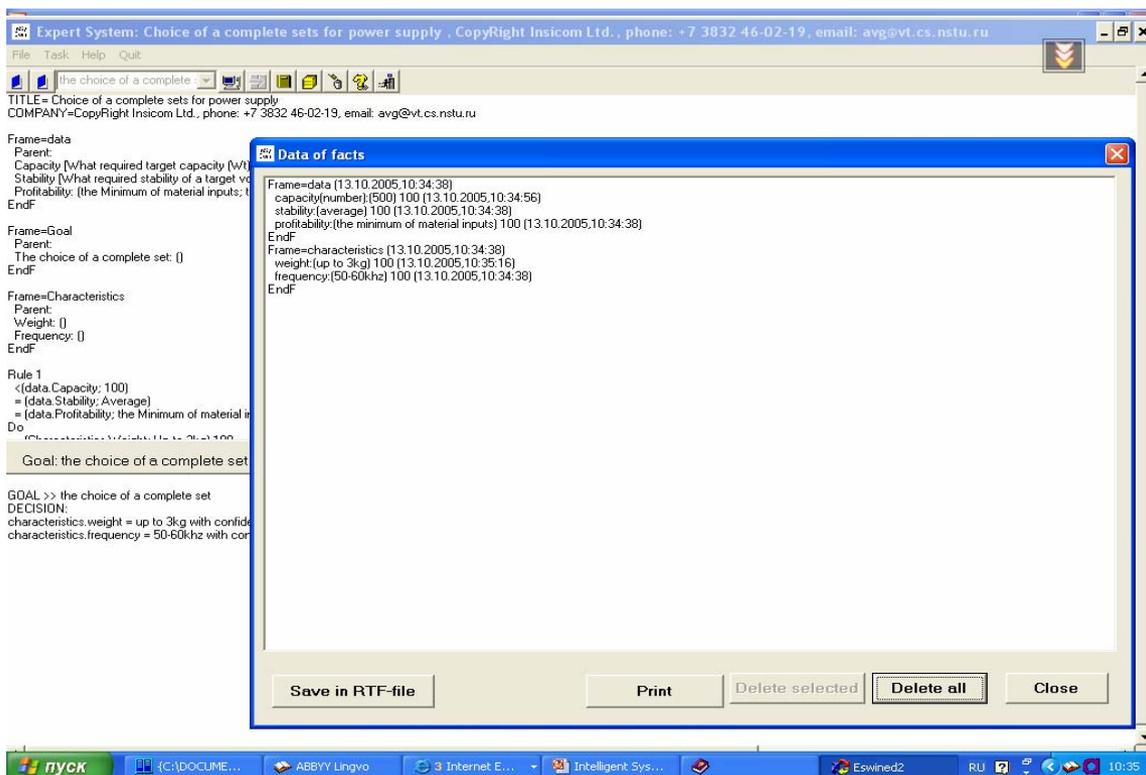


Figure 3. Screenshot of Expert Shell ESWin