BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **8**, No 2

Sofia • 2008

Soft Computing Agents in Browsing

Dimitar Lakov, Wu Zhimei** Shi Zhiqiang***

**Institute of Computer and Communication Systems, 1113 Sofia*
E-mail: lakov@iccs.bas.bg
*** Institute of Software – CA*
E-mail: szq@isdn.iscas.ac.cn

Abstract: *The paper applies a new flexible technique of Soft Computing Agents SCA in Internet browsing. It is called smart browsing since it takes into consideration the flexible defined client's preferences in the set of searching keywords. The initial point of browsing is a procedure of conventional Internet browsing. After receiving these results the browsing is repeated using soft computing technique. The browsing is applied treating all the entries in terms of fuzzy relations. Every entry is subject to analysis via fuzzy rule base individually defined by the client. As a result many of the inappropriate entries are truncated. SCA are used to perform this technique in remote data bases simultaneously. Then they feed back the results to the enquiring nod, where a new optimization is made applying the same technique. In respect to optimization SIMULINK of MATLAB and FuzzyTECH are applied. Concerning agent part of SCA a Bee-gent of Toshiba® tool is used to generate agent's kernels and disseminate them to the distant nodes. Some results on benchmark experiment are reported.*

Keywords: *Soft computing agents, Intelligent browsing.*

1. Introduction

The problems we intend to comment can be summarized as follows:

- Continuous growing information,
- Staled, old fashioned contaminated information,
- Excessive, irrelevant query interpretation.

At the US President's report addressing to Computational Science an information avalanche during 21st century [1] is pointed out. Estimated information growing in Internet is over one yonabyt (10^{18} bytes), but even in the last quarter of 2006 it exceeds 156 yonabyts. A special attention to the advanced methods for overall organization of information is proposed as an imperative measure. Basically there are three directions in optimization of processing into Internet:

(i) Organization of worldwide Internet browsers,

(ii) Semantic browsing,

(iii) Semantic questioning.

Concerning the first there is little hope to achieve some visible advance. The most powerful Internet systems such as Google, Yahoo, etc. have already reached their performance threshold. Every one of them supports a great net of thousands browser machines in extremely complicated organization. The only care is to assure mirroring for security considerations and fastest access time to prevent long queues and competitions. Even the fastest growing of processing principles fails to resolve the above mentioned problem, because it is not technological, but a paradigmatic one.

Semantic browser paradigm is a new direction that tries to resolve the problem, but it is not a panacea. A long craving human dream to bestow Internet with human assessable semantic is a matter of far future. The reason is that semantics is a notion that defies formalization even in human communications. To believe in its performance into Internet means to accept its organization as ‘clever’ as human beings are. Moreover if we accept this science fiction as reality there remains one more serious obstacle to be overcome. It is the compatibility. Due to the democratic principle of information repository building there is no way to escape from information contamination, old fashioned information, even spam generation. Every new semantic web organization meets the challenge of processing old information, that is impossible without authors. Such processing is neither possible nor necessary.

Our attention is attracted by some possibilities to combine contemporary Internet organization with a semantic assessment of common Internet user. It is not an artificial involvement of semantic into Internet abilities to interpret information, but improving the information retrieval, using the customer own semantics, which he undoubtedly possesses.

Our goal is to propose an advanced searching technique ready for immediate use. Since it is based on nowadays browsers facilities there is no need to change their organization. It relies on more active involvement of the human skills in interpretation of the searching process.

The paper is organized in four sections. The second chapter considers the application of Soft Computing Agents SCA as a perspective tool for intelligent browsing. Three soft computing models are considered. Among them Fuzzy Logic appears to be the most flexible and appropriate for immediate use. Simulations are discussed. In the fourth section a modelling of smart browsing is implemented using embedded Fuzzy Logic in Soft Computing Agents SCA. An experiment is discussed in the fourth section. The paper ends with a conclusion and a program for the next investigations in the field.

2. Smart Web Browsing

2.1. Presumptions

We call Smart Web Browsing (SWB***)*** a technique that uses SCA. The aim of SWB can be summarized in the following points:

- decreasing the size of the output of search information to a lightly conceivable list;
- involving of individual customer preferences into a set of input searching keywords in accordance with the fuzzy interpretation of the searching paradigm;
- applying a unified approach to the search for different type of the documents;
- improving the quality of searching due to lowering the size of the output list, and closer results to customer requirements.

The criterion for improving is the decreasing in the order of the output list size; as well the subjective customers' satisfaction in a search. There are two criteria for ranking the output list of searching:

- number of occurrences of the search pattern in data bases;
- number of links to the resultant page.

As you can see without some context in the searching procedure, the first has weak influence, whereas the second reflects better some other customer preferences or up to date interest about a concrete keyword. Involving customer's preferences by means of fuzzy interpretation of keywords we stress more on the personal individuality than on trends in global searching. [2] Moreover the last is not left out of searching attention since SWB is performed after conventional browsing.

2.2. Soft Computing agents

The information in Internet is miscellaneous by type, format, and presentation. Pertaining to the type, the Internet information is written, audible, visual, or combined of all the three types. Its format varies from .pdf to .html, .xtml and many others. As far as the presentation is concerned, it is up to all users without any constraints. In such rich variety and conditions the only way to process it is to mimic the natural human interpretation and dealing without any limitations, which can be posed by a formal mathematical representation. One way to do this is applying Soft Computing Technologies. There are many examples of useful application of Fuzzy Logic, Neural Networks, Evolution Algorithms, and others as a part of Soft Computing paradigm. [4, 6] In highly distributed Internet environment the application of Intelligent Agents become imperative. As a result of these two approaches the authors have involved a notion of SCA, which stands for a combination of both [5]. In what follows we consider three techniques incorporated in SCA.

2.3. Neural networks

At first in data mining applications we apply Normalized Radial Basis Neural Networks (RBNN) [7]. We suppose M pairs of input $(\mathbf{x})$ and output (y) data available in a searching process in every domain. The input vector **x** represents a term set of frequencies x_{i} of keywords: $\mathbf{x} = [x_1, ..., x_{\mathrm{i}}, \ldots, x_K]^{\mathrm{T}}$ with K being the index set of **x**. $\{x_1, y_1, \ldots, x_{\mathrm{j}}, y_{\mathrm{j}}, \ldots, x_M, y_M\}$ are a set of pairs with M being the number of entries in a domain containing at least one keyword. Every frequency of a keyword x_{ij} in vector **x** of user's defined input keywords corresponds to relative frequencies in j-th entry:

$$(1) \qquad x_{ij} = \frac{\boldsymbol{f}_{ij}}{\max\limits_{l} \boldsymbol{f}_{lj}},$$

$\boldsymbol{f}_{ij}$ is a real frequency in i-th keyword of j-th entry of d_j document, i.e. x_{ij} is normalized in respect to $\max\limits_{l} t_{lj}$ in j entry for $I = 1$ (l is the value for which x_{ij} is maximal). If a term does not appear in the d_j document, then x_{ij} is set to zero, y_{ij} is a single output presenting mean assessment of frequency in j-th particular searching result of K keywords, expressed by

$$(2) \qquad y_{ij} = \frac{\sum\limits_{i=1}^{K} x_{ij}}{K}.$$

Equation (1) permits relative comparison of different searching results in a semi-closed interval, (0, 1]; y_{ij} is also measured in a semi-closed interval, (0, 1]. Obviously it obtains the maximal value equal to one if and only if all x_j in a search entry j are equal. The sense is that the maximal assessment corresponds to even distribution of all keywords in a document regardless of their number. The task of a neural RBF is to present the grade of user's satisfaction. Then the modelled output y_{m} of the process, being within semi closed interval (0, 1], will be represented by a non-linear function

$$(3) \qquad y_m = f\,(\mathbf{x}),\ \mathbf{x} = [\,x_1, x_2, ..., x_K]^{\mathrm{T}}.$$

Here we accept Normalized Radial Basis Function Networks (NRBFN) similar to that accepted in [7]. NRBFN are able to represent the non-linearity. They consist of S RBF. The modelled output of the NRBFN is a weighted sum of the normalized outputs of all RBF***:***

$$(4) \qquad y_m(\mathbf{x}) = \sum_{i=1}^{S} w_i F_i\,(\mathbf{x}),$$

$$(5) \qquad F_i(x) = \mathrm{R}_i\,(\mathbf{x}) \Big/ \sum_{j=1}^{S} \mathrm{R}_j\,(\mathbf{x}),\ i=1,\ldots, S,$$

Equation (5) presents ***normalized strength*** of the i-th RBF, $R_i(\mathbf{x})$, i=1,…, S. The RBF is of ***Gaussian*** type that in a K-dimensional input space it is computed by the equation

$$(6) \qquad R_i(\mathbf{x}) = \prod_{j=1}^{K} \exp[-(x_j - c_{ij})^2 /(2\sigma_{ij}{}^2)] = \exp\left(-\sum_{j=1}^{K} [((x_j - c_{ij})^2 /(2\sigma_{ij}{}^2)] \right),$$

where i-th ***Gaussian*** function (i varies from 1 to S) is defined by the following vectors:

$\mathbf{c}_i = [c_{i1}, c_{i2},\ldots, c_{iK}]^{\mathrm{T}}$ presents the centre vector of neurons,
$\sigma_i = [\sigma_{i1}, \sigma_{i2}, \ldots, \sigma_{iK}]^{\mathrm{T}}$ is the width vector of their dispersion,
$\mathbf{w}_i = [w_{i1}, w_{i2},\ldots, w_{iK}]^{\mathrm{T}}$ is the connection vector of neurons.

NRBFN is defined by $2(S\times K)+S$ parameters. RBF learning is to adjust the three vectors $\mathbf{c}_i$, σ_i, and $\mathbf{w}_i$, so that to minimize Root Mean Square Error RMSE:

(7) $$\mathrm{RMSE}=\sqrt{\frac{1}{M}\sum_{i=1}^{M}(y_i - y_m^i)^2}\ .$$

An example of this technique is shown in Fig. 1.

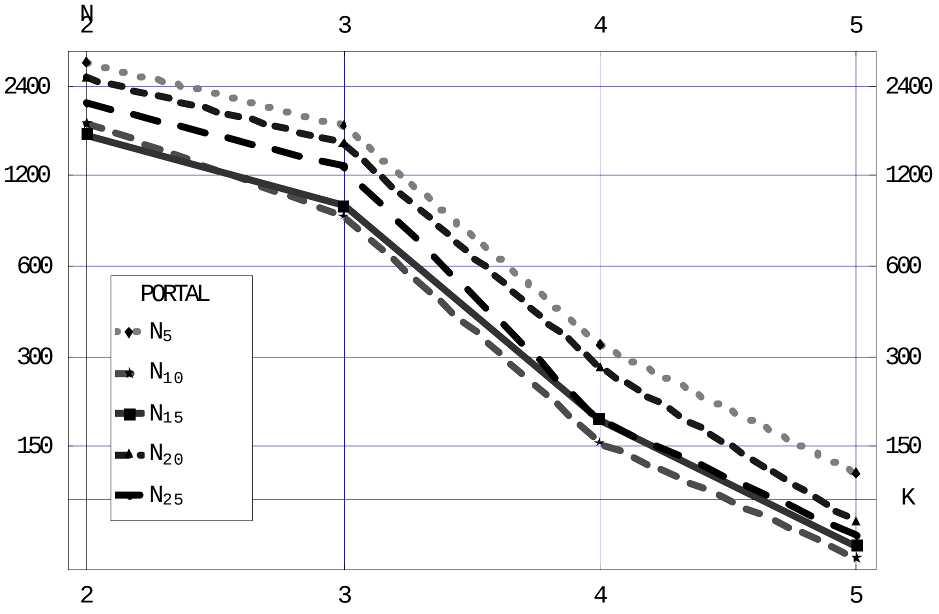

Fig. 1. Neural RBF Browsing

Application of NNRBF for data mining meets two challenges:

(i) On line learning of radial basis functions able to select prospective entities;

(ii) Distributed selection of entries based on calculated user's satisfaction $y_m(\mathbf{x})$ min.

The first one is optimized by a proper selection of tuning parameters in $2(S\times K)+S$ that minimize the learning trials. The fixed width of RBF decreases the parameters to $S\times K+S$ at the expense of slight deterioration in the output precision, which is not so significant for the browsing results.

The main problem of this browsing schema is the lack of objective criterion for tuning of N in NRBFN organization.

2.4. Immune algorithms

Artificial Immune Systems AIS are a selection technique that borrows some ideas from Self/Non-self Discrimination (SND). The SND mechanism allows AIS to discriminate cells in a similar to Living Immune System LIS way. Forrest et al [8] propose the Negative Selection (NS) algorithm based on this process. First, a set of detectors is generated; second, these detectors are compared against the self (normal); finally, those detectors that match any self element are discarded, others are kept. We resemble LIS discrimination to AIS partitioning in a data domain by the following way:

- selection of prospective entries in a given domain, based on a defined keyword set; they refer to genotype generation in LIS performed by conventional browsing;
- such obtained entries are divided into sub-sets containing really useful user's entities (self space), and others with shallow entities (non-self space);
- generation of 'intruder hunter' detectors that discard marginal entities in squeal.

Some scientists [8] have proposed fuzzy measure of abnormal deviation. They used a genetic algorithm with Deterministic Crowding. A fuzzy rule detector was defined as:

$$\text{(8)} \qquad \text{If } x_1 \in T_1 \wedge \text{ , …, } x_i \in T_i \wedge \text{ , …, } x_n \in T_n \text{ then non-self,}$$

where **x** = (x_1, …, x_n) is an element of the self/non-self space being evaluated; T_i is the fuzzy set defined by the union of some fuzzy sets in the fuzzy space of the attribute. A ***restricted-sum-or*** fuzzy logic operator and ***min-and*** was applied.

Encoding. Each atomic condition $x_i \in T_i$ is represented by a sequence ($s_1^i, \ldots, s_m^i$) of bits, where: $m = |S|$ is the size of the fuzzy space for the attributes set of linguistic values. The bit s_j^i is 'on' if and only if the corresponding basic fuzzy set S_j is a part of the composite fuzzy set T_i**.** Fig. 2 shows the structure of a chromosome.

s_1^1 ,…, s_m^1	,…,	s_1^i,…, s_m^i	,…,	s_1^n ,…, s_m^n

Fig. 2. Chromosome structure representing the condition part of a rule

Fitness Function. The fitness of a fuzzy rule detector R is calculated by (9):

$$\text{(9)} \qquad \text{fitness}(\mathbf{R}) = C(1 - \text{covering}(\mathbf{R})) + (1 - \text{C})\text{volume}(\mathbf{R})$$

$$\text{(10)} \qquad \text{covering}(\mathbf{R}) = \frac{\sum_{x \in Self} \text{eval}_{R(x)}}{|\text{Self}|}, \text{ volume}(\mathbf{R}) = \prod_{i=1}^{n} \text{measure}(T_i).$$

Covering**(R)** calculates the number of self samples being matched by the fuzzy rule detectors, measure ***(T_i)*** corresponds to the area under the membership function of the fuzzy set T_i**,** C **(**$0 \le C \le 1$**)** is a penalization coefficient for covering self samples – the higher coefficient the higher the penalization, typically between 0.8 and 0.9.

Distance Definition. We use the Hamming distance, representing each bit in the chromosome as a unique fuzzy set. An element **x** is classified as non-self (abnormal) with a Abnormality degree ($\mathrm{Ab}_{\mathbf{dg}}$) equal to the highest truth value of the fuzzy rule detectors evaluated on such element. A value of $\mathrm{Ab}_{\mathbf{dg}}$ close to zero means that **x** is normal and a value close to 1 indicates that it is abnormal. SND has the following steps:

- defining the Marginal Conditions MC of a non-self space of keywords to be searched. MC are calculated as a frequency of appearances;
- generation of detectors and testing them against entities containing keywords;
- discarding the entities that match to form output results of all domains.

An influence of abnormality degree into the searching process can be seen in Fig. 3. As the case in the previous section there are some difficulties with a natural tuning of system parameters. Besides, an immune optimization is not well defined in time. This can generate some inconvenience in its workability and application.

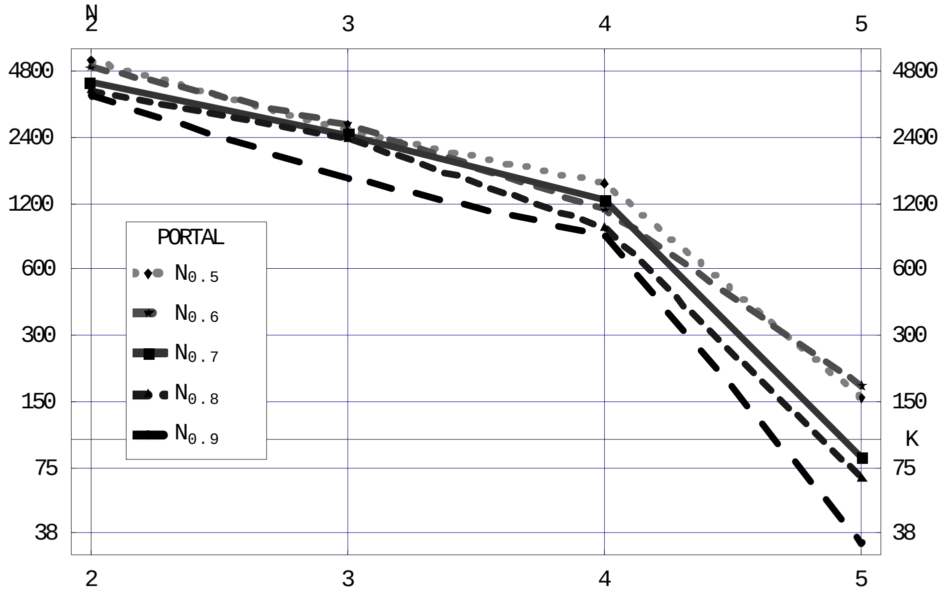

Fig. 3. Immune algorithm browsing

2.5. Fuzzy logic models

Fuzzy based inference aims to map a set of user defined keywords (term set) into assessment of quality of the search:

$$\boldsymbol{\beta}(x_j): \mathbf{X} \rightarrow \mathbf{Y}, \tag{11}$$

X is a set of searching vectors $\mathbf{x}_j$, $j = 1,\ldots, N$;

N – Number of entries containing at least one user's key word to be searched;

$\mathbf{x}_j = [x_1, ..., x_i ,\ldots, x_K]^{\mathrm{T}}$ is a vector defining user's preferences, $i = 1,\ldots, K$;

x_i is a keyword scalable within its linguistic terms;

Y is a set of linguistic fuzzy sub-sets reflecting quality of search within (0, 1];

An example of Fuzzy Inference Rule Base is of the form:

$$\begin{array}{c}
\text{If } x_1^1 \text{ is } x_{11}^1 \text{ and, \dots, } x_1^i \text{ is } x_{l1}^i \text{ and, ..., } x_1^K \text{ is } x_{L1}^K \text{ then } y_1^1 \text{ is } y_{11}^1 \\
\dots \\
\text{or } \boldsymbol{x_m^1} \text{ is } x_{1m}^1 \text{ and, ..., } x_m^i \text{ is } \boldsymbol{x_{lm}^1} \text{ and, ..., } x_1^K \text{ is } x_{Lm}^K \text{ then } y_m^1 \text{ is } y_{pm}^1 \\
\dots \\
\text{or } x_M^1 \text{ is } x_{1M}^1 \text{ and, ..., } x_M^i \text{ is } x_{lM}^i \text{ and, ..., } x_M^K \text{ is } x_{LM}^K \text{ then } y_M^1 \text{ is } y_{MP}^1
\end{array} \tag{12}$$

where:

K is a number of keywords (upper index of x_{lm}^i in (10));

l =1,…, L is a number of terms of antecedents, chosen to be equal for all keywords (first low index of x_{lm}^i);

m=1,…, M is a number of rules in fuzzy inference (second low index of x_{lm}^i in (12));

p =1,…, P is a number of terms of consequences (second low index of $\boldsymbol{y_{pm}^1}$ in (12));

For simplicity ‘and’ and ‘or’ are accepted to be min and max fuzzy operators; ‘and’, ‘or’ and ‘then’ assigning operators realize the so called real optimistic strategy of inference;

Representative membership functions are shown in Figs. 4 and 5 with three to five terms for antecedences and consequence. A fuzzy rule base is generated automatically as a hierarchical structure with complex rising hierarchy similar to that shown in Fig. 6. On its turn Fig. 5 shows fuzzy representation of output interpretations using five atomic terms. At a very initial stage Smart Web Browsing is modelled in FuzzyTECH environment using fuzzy inference technique for SCA. Three atomic fuzzy terms that present user’s requirements and literary ‘colour’ keywords have been chosen. To every keyword that is used in a conventional scheme of browsing machine one can attach an attribute. It expresses the individual preferences, which are strongly individual. They reflect the grade of confidence of client’s preferences to every keyword: ‘small’ – sm, ‘middle’– ml, and ‘large’ – lg. Obviously, the size of fuzzy rule inference data base depends on the number of keywords. For fully exhaustive rule base it spans from nine to 81 rules. The output of fuzzy inference system is a quality of assessment Q. It has five to nine atomic terms in rising complexity of fuzzy rule base. For the first they are: ‘Low’ – Lw, ‘low – Middle’ – l_M, ‘Middle’ – Md, ‘middle – High’ – m_H, and ‘High’ – Hg.

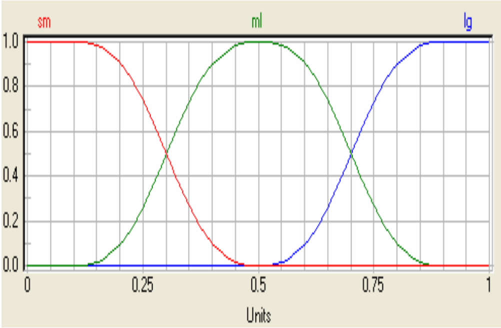

Fig. 4. Input terms

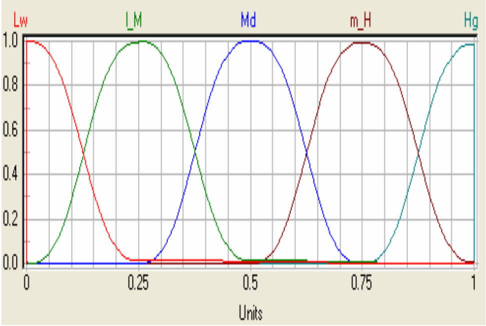

Fig. 5. Output terms

The representative fuzzy rule base scales *n* to *l*, where *n* is the number of input keywords. Obviously, for on line implementation, our goal is to minimize the fuzzy rule base in a situation where ***n*** is bigger than three. In such cases we decompose the initial fuzzy rule base into a number of hierarchical lower order sub-systems. In this case the initial fuzzy system is decomposed into two fuzzy sub-systems with their intermediate quality Q_1 and Q_2. As the second hierarchical level is the sub-system with two inputs: Q_1, Q_2 and one output Q – final combined quality. As you can see the initial complex system is decomposed into a finite number of low scale (up to three to one hierarchical fuzzy sub-system). Thus the maximal number of the fuzzy rules becomes 27. Similarly one can make a decomposition for another number of input keywords. Graphically the scheme of this decomposition of a five scale input keyword into two (two to one, and three to one) fuzzy sub-systems is given in Fig. 6.

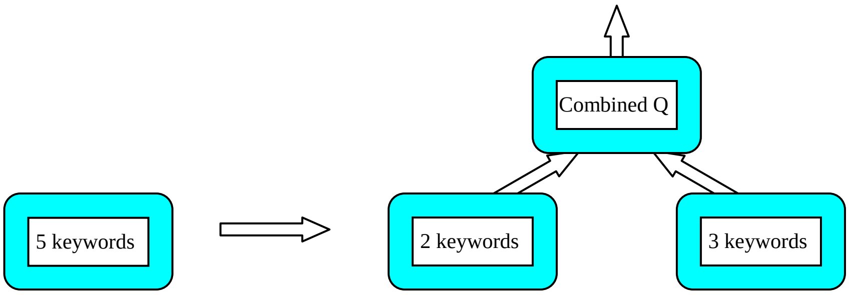

Fig. 6. Keyword decomposition

The other decompositions can be performed in accordance with Table 1. Principally, we gain in the application of very simple two and three order fuzzy systems.

Table 1. Decomposition of keyword fuzzy rule base

Number of input keywords	Number of sub-systems	Order of sub-systems
Four	two	second
Five	two	second and third
Six	two	two third
Seven	three	two second, one third
Eight	three	two third, one second
Nine	three	three third
etc.	…	…

The order of the sub-systems means the number of input keywords. Some explanation for using of Fuzzy Inference Rule Base is needed. In fact, the user does not need to understand in detail all the principles of its creation, but only a free logic of his natural querying. He involves his preferences in terms of defined fuzzy granules and defines them line by line. Besides there is no need to generate an exhaustive rule base. Practically the user spends a few minutes for combining his preferences in sentences, but the gain is in sequel decreasing of outgoing list of results.

Once a fuzzy rule base has been generated it forms a kernel of mobile SCA that is intended to perform a searching process in a remote node. From many existing mobile agent platforms we have chosen a Bee-gent of Toshiba® due to its clear entities definition and proven workability. On the other side the fuzzy model implementation at this preliminary stage is performed in MATLAB and FuzzyTECH environment for their good interpretation and visualisation. So, the first experiments are done using its agent platform and tools.

3. Modelling

A SIMULINK MATLAB modelling scheme of two hierarchical fuzzy rule data base is shown in Fig. 7. In accordance with Table 1 it represents five keywords. As an example here is presented a shoes searching process with five keywords: price, material, colour, size, and shoe number.

4. An experiment

The results of smart browsing over six portals are shown in Fig. 8. Google, AltaVista, AllTheWeb, Yahoo, HotBot and AllWet are presented. This initial browsing uses conventional technique of advanced Internet browsing.

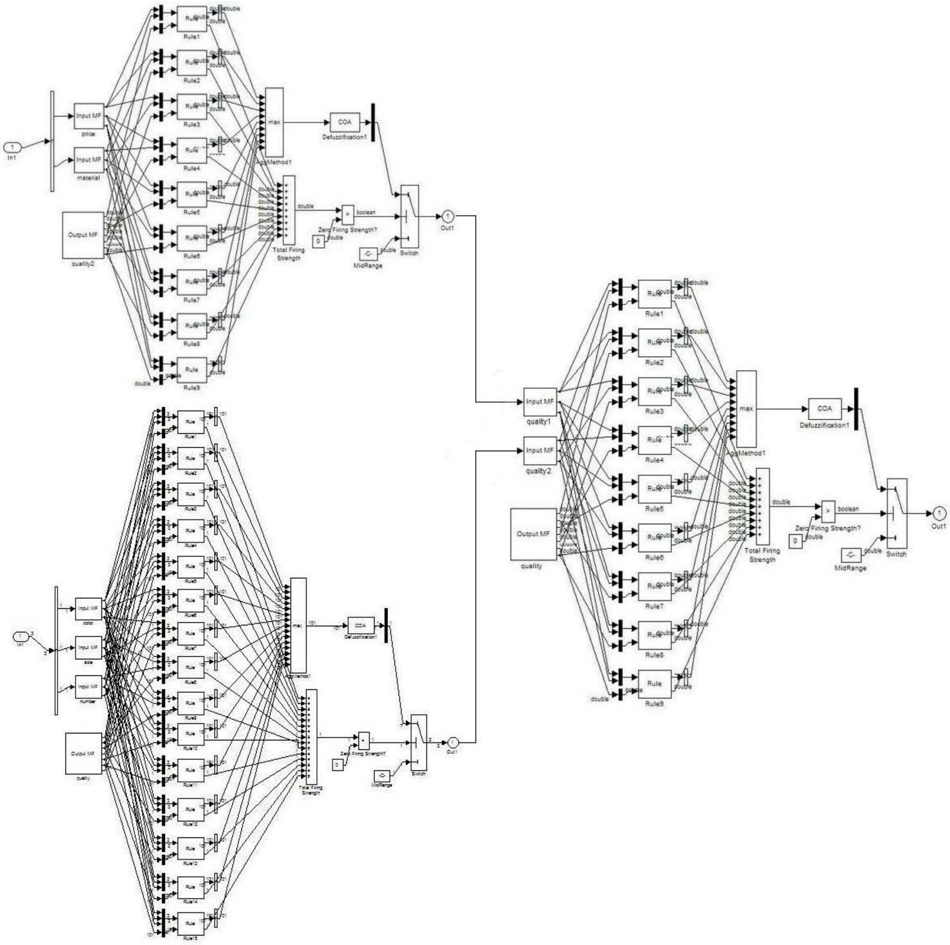

Fig. 7. Scheme of SIMULINK modelling

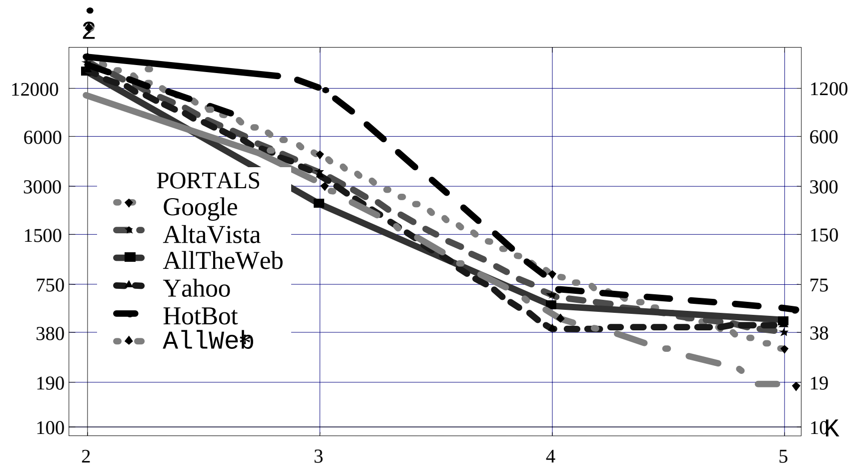

Fig. 8. Smart browsing reduction

All the experiments use the same benchmark in order to compare decisions of different portals in similar tasks. The left coordinate scales the results after conventional browsing. They are spanned between 20,000 and 50 entries depending on: the number of input keywords, type of the portal in use, and specificity of selected benchmark data base. As a representative benchmark an example commented in [2] is used. On the right side of the map a reduction after smart browsing is demonstrated. One can see that there is an order reduction in size of outgoing list. As one can expect the results of conventional browsing using different portals are similar. It is due to the similar principals that are used in advanced technique of browsing in all portals. Even the most advanced technique applied in all portals applies only Boolean principles of selection and scratching of inappropriate results. Using additional constraints such as time, format and others leads to some reduction, but risks in missing some important for the client results. In such a strategy the output relies more on the frequency of occurrences and links to the searching words than on the individual preferences of the client doing the search. This often does not reflect the real situation, but only the modern tendencies or the generalized estimation of the current state of Internet repository.

5. Conclusion

The paper demonstrates an approach in intelligent browsing technique, based on RBNN, AIA and fuzzy logic that are incorporated in Soft Computing Agents. The approach is not opposed to other intelligent methods developed within semantic WEB paradigm, but appears to be their competitor in this fast growing field. SWB is preceded by conventional browsing using well known and widely used portals such as Google, AltaVista, AllTheWeb, Yahoo, HotBot, and AllWet. The main idea behind such an approach is to obtain refinement of outgoing list based on the individual preferences of the client. They are involved by means of fuzzy defined attributes that 'coloured' the input keywords. The preliminary results show an order reduction in the size of the outgoing list. As a side effect comes the individual satisfaction of such browsing technique, which is strongly individual and has to be approved after profound investigations. The future investigations are envisaged in three directions:

- Using the rich possibilities of parallel browsing in remote data base nodes. So far the experiments consider conventional browsing of portals and mobile agents are used for definition of fuzzy rule data bases only,
- Although NNRBF and AIS methods show some inconvenience in their practical application, some of their ideas can be used for generation of new combined neuro/genetic/fuzzy algorithms for improvement of searching process.
- Applying profound investigations for comparison of the different Soft Computing methods in order to optimize the selection of the most appropriate one in concrete applications.
- Expected problems that have to be overcome:
- It is necessary to define clearly the range of smart browsing application. In some cases the client's enquiry does not allow fuzzy representation. A combined

technique of both conventional and smart browsing is imperative. In this case a browsing will be performed in one pass rather than using preliminary searching,

- Remote browsing requires another type of organization of indexed data bases of Internet repository. Some additional criteria have to be involved for smart selection,
- So far there exists a monopoly situation of main searching companies in managing of Internet repositories. An authorized, but free access to them would add more flexibility in solving the future problems in the fast growing information field.

References

1. Computational Science: Ensuring America's Competitiveness. President's Information Technology Advisory Committee, June 2005.
2. Zeldman, J. Designing with Web standards. Peachpit Press, July 2006.
3. Zadeh, L. A. Computing with Words and its Applications. – In: International Conf. on Soft Computing, Optimization and Manufacturing Systems, Miami, Florida, April 21-23, 2004.
4. Bradshaw, J. M. An Introduction to Software Agents. – In: Software Agents**,** J. M. Bradshaw, Ed., Menlo Park, CA, AAAI Press, 1997, 3-46.
5. Lakov, D. Soft Computing Agents. – In: Joint 9th IFSA World Congress/20th NAFIPS International Conference ISA/NAFIPS 2001, July 25-28, 2001, Vancouver, Canada, 2585-2590.
6. Loia, V., S. Sessa, Eds. Soft Computing Agents: New Trends for Designing Autonomous Systems, Studies in Fuzziness and Soft Computing. Physica-Verlag, Springer, Vol. **75**, 2001.
7. Vachkov, G. Classification of Machine Operations Based on Growing Neural Models and Fuzzy Decision. – ECMS 2007, Prague, Czech Republic, June 4-6, 2007, 68-73.
8. Forrest, S. et al. Self-Nonself Discrimination in a Computer. – In: Proc. of the IEEE Symposium on research in security and privacy, 1994, 46-54.
9. Kawamura, T. et al. Bee-gent: Bonding and Encapsulation Enhancement Agent Framework for Development of Distributed Systems. – In: Systems and Computers in Japan. John Wiley & Sons, Inc., Vol. **31**, 2000, No 13, 42-56.