2000

Information Filtering and Automatic Keyword Identification by Artificial Neural Networks

Zvi Boger
Nuclear Research Center, zvi@maxsw.com

Tsvi Kuflik
Ben-Gurion University of the Negev, tsvikak@bgumail.bgu.ac.il

Bracha Shapira
Rutgers University, b6480152@aol.com

Peretz Shoval
Ben-Gurion University of the Negev, peretz@bgumail.bgu.ac.il

search

Information Filtering and Automatic Keyword Identification by Artificial Neural Networks

Zvi Boger, Nuclear Research Center - Negev, and Optimal – Industrial Neural Systems Ltd., Beer-Sheva, Israel
zvi@maxsw.com

Tsvi Kuflik, Information Systems Program, Dept. of Industrial Engineering and Management, Ben-Gurion University, Beer-Sheva, Israel
tsvikak@bgumail.bgu.ac.il,

Bracha Shapira, Dept. of MSIS, School of Business, Rutgers University, New-Jersey,USA
B6480152@aol.com

Peretz Shoval
Information Systems Program, Dept. of Industrial Engineering and Management, Ben-Gurion University, Beer-Sheva, Israel
peretz@bgumail.bgu.ac.il.

Abstract

Information filtering (IF) systems usually filter data items by correlating a vector of terms (keywords) that represent the user profile with similar vectors of terms that represent the data items (e.g. documents). The terms that represent the data items can be determined by (human) experts (e.g. authors of documents) or by automatic indexing methods. In this study we employ an artificial neural-network (ANN) as an alternative method for both filtering and term selection, and compare its effectiveness to "traditional" methods. In an earlier study we developed and examined the performance of an IF system that employed content-based and stereotypic rule-based filtering methods, in the domain of e-mail messages. In this study we train a large-scale ANN-based filter which uses meaningful terms in the same database of e-mail messages as input, and use it to predict the relevancy of those messages.

Results of the study reveal that the ANN prediction of relevancy is very good, compared to the prediction of the IF system: correlation between the ANN prediction and the users' evaluation of message relevancy ranges between 0.76-0.99, compared to correlation in the range of 0.41-0.77 for the IF system. Moreover, we found very low correlation between the terms in the user profile (which were selected by the users) and the positive causal-index terms of the ANN (which indicate the important terms that appear in the messages). This indicates that the users under-estimate the importance of some terms, failing to include them in their profiles. This may explain the rather low prediction accuracy of the IF system that is based on user-generated profiles.

1. Introduction

Information Filtering (IF) is a research area that offers tools for discriminating between relevant and irrelevant information. It provides personalized assistance for continuous retrieval of information in situations of information-overflow in general, and on the Internet in particular. Information filtering combines together tools from the field of artificial intelligence (AI), such as intelligent agents or software robots ("softbots"), guided by user profiles, with information retrieval (IR) methods, geared to representing, indexing and retrieving of content [13,14, 4]. IF differs from traditional IR in that it is dealing with users who have long term interests (information needs) that are expressed by means of user profiles, rather than casual users whose needs are expressed as ad-hoc queries [5].

Agent technology provides a framework for automated information gathering over the Internet. Indeed, quite a few applications have been developed for this purpose. Passive filtering of incoming messages, like email and Usenet data, presents one such application [25]. Active information seeking, like interesting web-site detection and browsing assistance, presents another application [19,4,23,24]. The heart of such an agent is "user profile"; a representation of user needs which is being constantly updated according to user feedback. The performance of IR and IF systems (namely their ability to retrieve or filter relevant information) depends on many factors, one of that is the selection of keywords that represent the user query or profile.

Artificial neural networks (ANN) are used in recent years for modeling complex systems where no explicit equations are known, or the equations are too ideal to represent the real world. The ANN can form predictive models from data available from past history. Advanced algorithms can train large ANN models, with thousands of inputs and outputs. Analysis of the trained ANN may extract useful knowledge from it.

Keyword selection for the specification of user profiles or queries is an important, and sometimes frustrating, task. In this paper we try to handle this task by training a large-scale ANN-based filter which uses all meaningful words in the document space (i.e. data items) as inputs. Analysis of the trained ANN may achieve automatically important keyword identification. To test this technique we use a rather small user-ranked database of e-mail messages that was compiled for testing filtering methods by statistical techniques.

The rest of this paper is structured as follows: Section 2 reviews some essential concepts in IR and IF. Section 3 provides a brief introduction on ANN modeling techniques, and Section 4 describes possible application of ANN to information filtering. In Section 5 we present the results of a prior study on filtering e-mail messages, which combines content-based and rule-based sociological filtering, integrated with user stereotypes. This lays the ground for our implementation of the ANN approach to the same data (e-mail messages): Section 6 explains the ANN model building and compares the results of the ANN approach as an information filter to the results of that earlier study. Section 7 analyzes the ANN prediction of the importance of words in text, and compares it to user evaluation of term importance. Section 8 concludes and discusses further research issues.

2. Concepts in Information Retrieval and Filtering

Information retrieval (IR) may be characterized as "leading the user to those documents that will best enable him/her to satisfy his/her need for information" [28]. This definition (among many others) can be described in a model of information retrieval where the user seeks, by the use of queries, relevant information in some data space (e.g. a database of documents). Years of research in IR yielded many useful results, among them document representing and indexing methods.

Interesting lessons learned from IR are in three main areas: text representation, retrieval techniques, and acquisition of user information needs. The vector space model [30], according to which a document is represented by a (possibly weighted) vector of terms, is perhaps the most commonly used model for text representation. The user information interests (i.e. queries) can be represented as a vector of keywords in a similar way [1,5,26]. The main task of IR, given user queries and data representation, is to match the two vectors of terms and thus provide the user with relevant data items that best match the query.

There are several different IR models for determination the weights of term in documents or queries. The Boolean model assigns equal weights to all terms in the query, and thus matching is based simply on the appearance of those terms in the data items (satisfying the Boolean logic conditions of the query). In statistical models, weights reflect the importance of terms. One well-known method for determination of term weights in documents is TFD*IF (Term Frequency * Inverse Document Frequency) [30]. It assigns a weight to a term in proportion to the number of its occurrences in the document and in inverse proportion to the number of documents in which it occurs at least once. This method is based on the statistical observation that the more times a term appears in a text the more relevant is the topic, and the more documents it appears in, the more poorly it discriminates between documents. Another model for determination of term weights is a probabilistic model that uses the difference in the distribution behavior of words over all documents in a collection to guide the selection of index terms.

In IF systems user needs are expressed as profiles. A profile represents his/her long-term information needs. There are two main distinct user profile approaches [26, 1,3]:

- One) Content based profile: represents the user's areas of interest by a set of terms. The profile can be defined "manually" (i.e. provided by the user), or generated automatically from a sample set of data items that are known to be of interest to the user.
- Two) Collaborative profile: this approach matches user's information rating patterns. The main assumption is that people with similar rating patterns seem to like the same kind of information. Therefore, it makes sense to offer them information that was liked by people with similar rating pattern – "like minded people".

In order to support user needs, the user profile should be adaptable according to feedback from user reaction to information provided to him/her, since user interests tend to change over time. This calls for incorporating learning mechanisms into user profiling [6]. A learning mechanism that suites user profiling is inductive learning: An initial profile is generated (either manually or automatically, as explained), following some initial training from examples, and then additional learning is done according to user feedback, in order to continuously improve the user profile.

3. Brief Introduction to Artificial Neural Networks Modeling

ANN modeling is done by learning from known examples. A network of simple mathematical "neurons" is connected by weights. Adjusting the weights between the "neurons" does the training of the ANN. Two main branches of ANN are in use, separated by their training methods: supervised and unsupervised:

- One) The supervised ANN branch uses a "teacher" to train the model, where an error is defined between the model outputs and the known outputs. Error back-propagation algorithm adjust the model connection weights to decrease the error, by repeated presentations of inputs vectors [29]. The most used ANN structure is the fully connected feed-forward, with one hidden layer of sigmoid activation function neurons.
- Two) The unsupervised ANN branch tries to find clusters of similar inputs when no previous knowledge exists about the number of the desired classes. The most

known algorithms are the Self-Organized Map (SOM) and Adaptive Resonance Theory (ART).

In both cases, once the ANN is trained, and verified by presenting inputs not used in the training, the ANN is used to predict outputs of new inputs presented to it. (The reader is referred to the many books and journal papers published on these subjects. An example of SOM application to large database similarity finding can be found in [20, 15]. An ART application can be found in [18], while recent feed-forward ANN examples can be found in [12, 21]).

There are several obstacles in applying ANN to large systems containing large number of inputs and outputs. Most ANN training algorithms need thousands of repeated presentations ("epochs") of the inputs to finally achieve small modeling errors. Large ANN tends to get stuck in local minima during the training. As most ANN training start from initial random connection weights sets, and the number of neurons in the hidden layer are usually determined by heuristic rules, many re-training trials are needed to achieve good models.

The PCA-CG training algorithm [17] can easily train large scale ANN models, as it pre-computes non-random initial connection weights from the manipulation of training data sets. A proprietary algorithm avoids, and escapes, local minima. This algorithm was successfully used to train ANN models of industrial plants with hundreds of inputs and outputs, [7,8]. It was also used for spectra and image analysis [10,11,22,16].

Once trained, the ANN may be analyzed for knowledge extraction. One way is to estimate the relationships between the inputs and the outputs. A simple algorithm can calculate a Causal Index (CI) that gives the relative magnitude and the sign of the influence of each input on each output [2]. The CI is calculated as the sum of the product of all "pathways" between each input to each output,

where there are h hidden neurons, W_{kj} are the connection

$$CI = \sum_{j=1}^{h} Wkj * Wji$$

weights from hidden neuron j to output k, W_{ji} are the connection weights between input i to hidden neuron j. Although there is no rigorous theoretical basis for this algorithm, experience with large scale plant modeling shows that the CI found by this algorithm do indicate the known (and sometimes previously unknown) relationships in the data.

Another knowledge extraction technique is the ranking of the inputs according to their relevance to the ANN prediction accuracy [9]. The least relevant inputs may be discarded and the ANN re-trained with the reduced input set to give better prediction accuracy. The explanations for this possible improvement are: a) The elimination of noise or conflicting data in the non-relevant inputs. b) Reduction of the number of connection weights in the ANN, that improves the ratio of the number of examples to the number of connection weights, thus reducing the chance of over-fitting.

4. The Application of ANN to Information Filtering

The idea to match the capabilities of ANN modeling to information retrieval is not new. A search of the 1994-1998 INSPEC database with these terms yielded more than 60 papers dealing with this combination. Most of the papers use the SOM or the ART techniques to form clusters of textual documents, based on the similarity of the keywords in the texts. Once trained, the ANN will classify new documents as belonging to one of these clusters.

The ability of the ANN to model non-linear, non-obvious relationships can be applied to the matching of the textual features (inputs to the ANN) to the user profile (ANN outputs). In contrast with the statistical methods used for the required modeling, no assumptions need to be made (such as assuming normal distribution, subjectively selecting the number of terms, and the form of the model equations).

Of the two ANN training methods, the supervised training should be preferred, as it is more adjustable to an individual user profile. The SOM may classify text according to their similarity, but eventually the user will have to evaluate the number of clusters (too few or too numerous), and to rank the clusters according to their degree of interest.

The most important feature of the ANN modeling is that the user need not specify what features to extract from the text, such as selecting the keywords. If the ANN can use all the words in the text as inputs, the post-training ANN analysis should reveal what are the more relevant words in the text, according to the user profile. Thus the subjective keywords selection process for the queries is avoided, eliminating the frustration of getting too many responses to a general query, or the suspicion of missing important results from a too narrow selection of keywords. The trained ANN should then act as a filter, evaluating each additional text according to the ANN predicted match to the users profile requirements.

In this paper we try to prove the feasibility of ANN modeling and keyword extraction, employing a database used in a previous modeling of user's profiles by other techniques.

5. Information Filtering utilizing Content-based and Rule-Based Methods

Shapira *et al.* [31] developed a dual-method model and system for filtering and ranking relevance of information. One method is "content-based" filtering, which is based on the correlation of two weighted vectors of terms, one representing the user profile and the other representing the data items. The other method is "sociological filtering" integrated with user stereotypes. A user stereotype is a common place for users who have common information usage and filtering behavior, expressed by a set of filtering rules. Each filtering rule, if found relevant for a data item being evaluated, grants a relevance value to that item. The overall "sociological relevance" of the data item is the average relevance values of those rules. When the filtering system evaluates a data item for a user, it first identifies that user's stereotype (based on the similarity of the user's sociological profile to the respective profiles that represent each of the stereotypes). Then the filtering rules of that stereotype are applied, and the relevance of the data item is calculated, as described above.

A prototype system was developed to test the applicability of the model for filtering e-mail messages, and experiments were run to determine the effects of combining the two filtering methods in various filtering strategies. Ten users, university people, were asked to evaluate e-mail messages that they received from list servers they subscribed to, and to rank their relevancy on a 7-point scale. The content-based profile of each user was determined as follows: Each of the ten participants received a proposed list of terms that was generated from a "training set" consisting of several dozen of his/her incoming e-mail messages. The list included the most frequently occurring terms in those messages. (It was prepared with the aid of special software that extracts meaningful terms from messages, employing stemming algorithm, look-up tables and a stop-list, and counts the frequency of the meaningful terms). Each participant was asked to review the proposed list of terms, add or drop terms, and weigh each term for its degree of interest to him, using a 0-100 scale. Then, the system evaluated the same messages several times, each time employing a different filtering strategy, namely: content-based alone; sociological filtering alone; both methods (parallel), where the final relevance rank is the average of both methods; content-based followed by sociological; and sociological followed by content-based filtering. In each of the last two cases the first (primary) method contributed 70% of the final relevance rank. In all cases the relevance rank of the system was expressed on a 7-point scale, as used for user evaluations, to enable comparisons of results.

To enable content-based filtering, the system analyzed each message by stemming the words, eliminating "stop-list" terms and counting the frequency of the meaningful stems, thus generating a weighted vector of terms, that was correlated with the user's content-based profile. To enable sociological filtering, the system used the same word stems and employed specific algorithms attached to each rule to determine if and how relevant is the rule to the message. Based on that it calculated the relevance rank for each rule and computed the overall sociological rank.

The performance of the system (i.e. the relevance ranks determined by the system), according to those various strategies, was compared to the user evaluations. Some relevant results of the experiments are summarized in Table 1. Each column refers to one of 4 stereotypes determined for the user population in that experiment. Each row shows the correlation (r) between the system's rankings of the messages and the users' rankings. (N is the number of messages evaluated by users within each stereotype.)

Table 1. Correlation (r) Results per Stereotypes

Filtering Strategy	Stereotype 1 (N=429)	Stereotype 2 (N=469)	Stereotype 3 (N=179)	Stereotype 4 (N=350)
Content-based alone	.58	.50	.47	.41
Sociological alone	.48	.48	.43	.65
Parallel (both)	.59	.64	.59	.71
Content-based (70%) + Sociological (30%)	.61	.44	.62	.53
Sociological (70%) + Content-based (30%)	.52	.07	.51	.66

The results reveal that content-based filtering alone is usually more effective than sociological filtering alone, but that combinations of both methods yield better results than each method individually. The best filtering strategies are achieved when the two methods are used in parallel, or when content-based filtering is the primary method, followed by sociological filtering. At any rate, correlation between system predictions and user evaluations are usually not very high, ranging (with one exception) between 0.4-0 .7.

6. Training the ANN as Information Filter

The gathering and pre-processing of the training and testing data is the first phase (and frequently, the most time-consuming phase) of ANN modeling. In our case most of the data was already collected and classified as described in the previous section - we had the ten e-mail recipients' classification of the importance of the messages they received, on a scale of 1-7. The words in their messages were already stemmed, and common words removed by a stop-list. We had also the content-based profiles, i.e. the users'-given weights of each word as a subjective measure of relevance, on a scale of 0-100.

Altogether 1524 e-mail messages were used to form 10 user profiles, aiming to filter future e-mail messages according to their importance to the recipient.

In our case, the aim was twofold: a) to predict relevancy of the messages, and b) to evaluate the ability of the ANN to identify important keywords for the user profile. All the stemmed words in the messages were combined into one "keyword" list, 425 in length. Each e-mail message was transformed into a binary vector of 425 ones and zeros, the ones signifying the presence of the indicated word in the message. The data preprocessing to the form used in the ANN training consisted of changing the one and zeros binary inputs to +1 and -1 values, respectively, and adding a small random noise value to them. This was done in order to avoid having an empty input column vector of constant -1 values. The 1-7 output range was transformed into the usual 0.1-0.9 range, which avoids asymptotic numerical problem during the training. As the aim of this paper was to compare the ANN modeling with the stereotype method results, all examples were used for the training, (as was done previously). Thus no validation set was set aside to check the generalization capacity, as is usually done in ANN model development.

Ten fully connected ANN models were trained, one for each user. The models were trained with all available 425 wide vectors, with the 1-7 rating as a single output. Normally some of the available examples are not used in the training, but are used to check the generalization capacity of the trained ANN. The training algorithm suggested the number of neurons in the hidden layer, **h**. It was equal to 6 at most, sufficient to achieve good prediction rates. The small number of hidden neurons, compared with most other published ANN models, is an important feature of the ANN algorithm we use.

Some of the ANN models (users 6, 8, 9) could not be trained to the desired accuracy. The "imperfect" ANN models were analyzed to identify the more relevant inputs, as described in [9]. These ANN models were retrained with the reduced keyword sets to give high model accuracy, as described at the end of section 3. Table 2 summarizes the results of the ANN approach, compared to the results of the "traditional" IF approaches. As can be seen, the ANN prediction accuracy is very good, ranging between 0.76-0.99, while the "traditional" methods yielded results of up to 0.79 success at the most. These good results are expected: ANN modeling should improve the filtering accuracy compared with linear modeling, as the number of adjustable connection weights in the ANN is, in this case, higher than the number of examples.

Table 2. Prediction Accuracy of ANN and "traditional" Filtering Methods (Correlation between model prediction and user evaluation)

User	ANN	Content-based	Sociological	Parallel (both)	Sociological (70%) + Content-based (30%)	Content-based (70%) + Sociological (30%)
1	0.96	0.60	0.42	0.53	0.56	0.46
2	0.90	0.43	0.31	0.48	0.40	0.31
3	0.88	0.44	0.38	0.47	0.44	0.41
4	0.80	0.25	0.47	0.51	0.36	0.47
5	0.76	0.31	0.53	0.56	0.27	0.55
6	0.99	0.64	0.61	0.72	0.71	0.66
7	0.93	0.23	0.46	0.52	0.46	0.51
8	0.97	0.59	0.61	0.70	0.42	0.63
9	0.98	0.73	0.54	0.68	0.79	0.59
10	0.99	0.62	0.65	0.78	0.74	0.77

7. ANN as Predictor of Term Importance

The next stage was to analyze the trained ANN to learn more about the real importance of the words in the e-mail message. The Causal Index of each model was calculated, as described in Section. 3. The magnitude relative to the other calculated pathways and the sign of the summed products may be interpreted as the overall degree and direction of the influence of a particular input on a particular output. In our case, large positive CI means that the presence of this keyword in message tends to increase it's relevance. CI close to zero means that the keyword cannot be used for classifying the message relevance. The meaning of large negative CI is that messages containing this keyword tend to be less relevant. However, we do not analyze its meaning in this context; we refer only to the large positive CI keywords.

As described in Section 5, the users gave their estimate of the importance of each keyword, on a scale of 0 – 100, to construct their content-based profiles. One possible explanation for the low accuracy of the previous models' prediction is that the users do not correctly identify the relative importance of the keywords to be included in their profiles. Thus a comparison between the positive CI and the user given importance rating to these words should be interesting. As spurious small CI may result from the small random noise addition to the inputs, only CI values whose magnitude was larger than 0.1 were used. The results are given in Table 3, using the Pearson statistic to find a correlation.

Table 3: Correlation between Users' Rated Importance and the CI

User	1	2	3	4	5	6	7	8	9	10
Pearson	0.17	0.19	0.07	0.20	0.15	-0.02	0.20	0.05	0.35	0.09

As can be seen from that comparison, no correlation was found. This suggested that the users' subjective importance ratings (i.e., the user-defined content-based profiles) were not good enough, or the CI method is faulty.

To test which hypotheses is correct, we calculated the mean *user-given* importance rating of the e-mails in which words with positive CI values had a *zero* importance user rating. That is, those words that the users believed had no importance for them. For each ANN based user profile we randomly selected a list of 10 terms which had positive CI value but the user assigned 0 importance for the content-based profile. For comparison, we show also the overall mean rating of each user's e-mail messages. The results are given in Table 4.

Table 4: Mean Importance Rating of Positive CI Keywords with Zero Rating by Users

User	1	2	3	4	5	6	7	8	9	10
ANN	3.54	3.36	5.01	3.78	3.04	4.81	5.33	4.30	4.24	5.33
Overall	2.74	2.73	3.41	3.04	2.29	4.03	3.88	3.07	3.78	4.8

As can be seen from Table 4, the keywords that the ANN CI identified as important, show up in e-mail messages whose importance rating is higher than the overall mean. Thus it seems that users under-estimate the importance of some of the words in the e-mail message, which may explain the rather low prediction accuracy based on the users' subjective rating.

8. Conclusions and Suggestions for Further Research

The results presented in Sections 6 and 7 show that large ANN model can successfully be trained from the non-trivial words in a text, and give better prediction than statistically derived models. The ANN can be analyzed by the CI method to identify the more important keywords. The users'-given rating of the keywords may be too subjective, or given without too much attention. It would be interesting to receive their evaluation of the CI generated keyword importance. It has to be kept in mind, though, that the database is rather statistically small, and may be the 1-7 rating is too detailed for exact classification by users.

Some of the usability features of the new ANN-based model need further study and will be the subjects of further research. One of them is the number of words in the user vocabulary, as related to the ability to train large scale ANN models. ANNs with several thousands inputs were successfully trained with the CG-PCA algorithm [8]. It is only a matter of large enough computer memory, and training time. However, once a large ANN is trained, additional retraining based on the ANN trained on an existing list of words is an easy task, if enough blank columns are used in the original training. To reduce the number of words in the user vocabulary it will be necessary to use subject specific thesauri, so that a new word will be checked to find if it can be represented by a synonymous word in the current ANN vocabulary.

Other usability issues are: the number of training data needed to train a good predictive ANN; the need of periodical re-training to include the new words and the user evaluation of new incoming e-mail messages.

An important feature is the ability to predict correctly the relevancy of a data item that contains new words. One approach is to calculate some metric that will generate "not sure" warning alongside the ANN prediction. This warning will prompt the user to read this data item and judge its relevancy in the next updating of the ANN. Such metric may be based on the average (absolute) Causal Index values of all the words in the database (with the new words, having a zero CI, lowering the result). A "not sure" warning value will be generated for items having an average CI below some threshold.

References

[1] Aas, K. *A Survey on Personalized Information Filtering Systems for the World Wide Web*. 1997, Report No. 922, Norwegian Computing Center.

[2] Baba, K., Enbutu, I. and Yoda, M. "Explicit Representation of Knowledge Acquired from Plant Historical Data Using Neural Network," *Proc. Int. Joint Conf. on Neural Networks. San Diego*, 1990, (3), 155-160.

[3] Balabanovic', M., Shoham, Y. "Learning Information Retrieval Agents: Experiments with Automated Web Browsing". *Spring Symposium on Information Gathering from Heterogeneous Distributed Environments.* AAAI 95, 1995.

[4] Balabanovic', M., Shoham, Y. "Fab: Content-Based, Collaborative Recommendation". *Comm. of the ACM*, 1997, 40 (3), 66-72.

[5] Belkin, N. J. and Croft, W. B. "Information Filtering and Information Retrieval: Two Sides of the Same Coin?" *Comm. of the ACM*, 1992, 35 (12), 29-38.

[6] Billsus, D. and Pazzani, M. "Learning Collaborative Information Filters". *Proc. of the Int'l Conf. on Machine Learning*, 1998, Morgan Kaufman Pub., Madison, Wisconsin.

[7] Boger Z. "Application of Neural Networks to Water and Wastewater Treatment Plant Operation". *Trans. of the Instrument Society of America*, 1992, 31 (1), 25-33.

[8] Boger Z. "Experience in Industrial Plant Model Development using Large Scale Artificial Neural Networks". *Information Sciences – Applications,* 1997, 101 (3/4), 203-215.

[9] Boger, Z., and Guterman, H. "Knowledge Extraction from Artificial Neural Networks Models". *Proc. of the IEEE Int'l Conf. on Systems Man and Cybernetics, SMC'97*, Orlando, Florida, 1997, 3030-3035.

search

[10] Boger, Z. and Karpas, Z. "Application of Neural Networks for Interpretation of Ion Mobility and X-Ray Fluorescence Spectra". *Analytica Chimica Acta*, 292, 1994, 243-251.

[11] Boger, Z. and Karpas, Z. "Use of Neural Networks for Quantitative Ion Mobility Spectrometric Measurements". *J. Chem. Info. Comput. Sci*., 34, 1994, 576-580.

[12] Creput, J-C. and Caron, A. "An Information Retrieval system using a New Neural Network Model". *Cybernetica*, Vol. XL 2, 1997, 127-139.

[13] Etzioni, O. and Weld, D. "A Softbot-Based Interface to the Internet". *Comm. of the ACM*, 1994, 37 (7), 72-76.

[14] Etzioni, O. and Weld, D. "Intelligent Agents on the Internet: Fact, Fiction and Forecast". *IEEE Expert*, 1995, 10 (4), 44-49.

[15] Graupe, D. and Kordylewski, H. "A Large Memory Storage and Retrieval Neural Network for Adaptive Retrieval and Diagnosis". *Int'l J. of Software Engineering and Knowledge Engineering*. 1998 (6), 115-138.

[16] Greenberg, S. and Guterman, H. "Neural Networks Classifiers for Automatic Real-World Image Recognition". *Applied Optics*, 1996, 35, 4598-4609.

[17] Guterman, H. "Application of Principal Component Analysis to the Design of Neural Networks". *Neural, Parallel and Scientific Computing*, 1994, 2, 43-54.

[18] Hui, S.C. and Lau. K.L. "An Application of Neural Networks in Document Retrieval". *The New Review of Applied Expert Systems,* 1997, (3), 69-80.

[19] Joachims, T., Freitag, D. and Mitchell, T. "A Tour Guide for the World Wide Web". *Proceedings of IJCAI97*, 1997.

[20] Kohonen, T. "Exploration of Very Large Databases by Self-Organizing Maps". *Proc. of the IEEE Int'l Conf. on Neural Networks*, 1997, 1, PL1-6.

[21] Kurbel, K., Singh, K. and Teutenberg, F. "Search and Classification of 'Interesting' Business Applications in the World Wide Web using a Neural Network Approach". In Forcht, K. (Ed.), *Proceedings of the 1998 IACIS Conference in Cancun*, Mexico, Oct. 1-3, 1998, 75-81.

[22] Lerner, B., Guterman, H., Dinstein, I. and Ronen, Y. "Human Chromosome Classification using Multilayer Perceptron Neural Network", *Int'l J. Neural Systems*, 1995, 6, 359-370.

[23] Lieberman, H. "Letizia: An Agent that Assist Web Browsing". *Proc. of the Int'l Joint Conference on Artificial Intelligence*, Montreal, 1995.

[24] Lieberman, H. "Autonomous Interface Agents". *Proc. of the ACM conference on Computers and Human Interface, CHI 97*, Atlanta, Georgia, 1997.

[25] Maes, P. "Agents that Reduce Work and Information Overload". *Comm. of the ACM*, 1994, 37 (7), 31-40.

[26] Oard, D., W., Marchionini, G. "A Conceptual Framework for Text Filtering". *Technical Report* EE-TR-96-25 CAR-TR-830 CLIS-TR-96-02 CS-TR-3643, 1996.

[27] Pazzani, M. J. "A Framework for Collaborative, Content-Based and Demographic Filtering", *AI Review (in press)*; www.ics.uci.edu/~pazzani/publications/aireview.html

[28] Robertson, S. E. "The Methodology of Information Retrieval Experiment". In Sparks Jones K. (Ed), *Information Retrieval Experiment*, Butterworths, 1981. Ch. 1, 9-31.

[29] Rumelhart, D. E., Hinton, G. E. and Williams, R. J. "Learning Representations by Back-Propagating Errors", *Nature*, 1986, 323, 533-536.

[30] Salton, G., McGill, W. J., *Introduction to Modern Information Retrieval*, McGraw-Hill. New York, 1983.

[31] Shapira, B., Shoval, P. and Hanani, U. "Experimentation with an Information Filtering System that Combines Cognitive and Sociological Filtering Integrated with User Stereotypes", *Decision Support Systems*, 1999 (forthcoming).