SIGMA: Integrating Learning Techniques in Computational Markets for Information Filtering

Ferguson, I.A.; Karakoulas, G.J.

SIGMA: Integrating Learning Techniques in Computational Markets for Information Filtering

Grigoris J. Karakoulas

Global Analytics Group
Canadian Imperial Bank of Commerce
161 Bay St., BCE-8, P.O. Box 500,
Toronto ON, Canada M5J 2S8
karakoul@cibc.ca

Innes A. Ferguson

Interactive Information Group
Institute for Information Technology
National Research Council,
Ottawa ON, Canada K1A 0R6
innes@ai.iit.nrc.ca

Abstract

This paper presents an adaptive model for multi-agent learning based on the metaphor of economic markets, that can cope with the non-stationary and partially observable nature of an information filtering task. Various learning and adaptation techniques – i.e. reinforcement learning, bidding price adjustment and relevance feedback – are integrated into the model. As a result of this integration learning through the model exploits market competition in order to dynamically construct mixtures of 'local experts' from selfish agents. The model is embedded into SIGMA (System of Information Gathering Market-based Agents) for information filtering of Usenet netnews. The functionality of the system is discussed together with work underway for its evaluation.

1. Introduction

There has been a growing interest in employing machine learning techniques in information filtering (IF) due, in part, to the capabilities of such techniques to deal with multi-dimensional, partially structured, and noisy data. The latter characterize, on one hand, information services such as Usenet, newswire feeds, the World Wide Web and repositories of electronic documents; and on the other, users' interests that need to be mapped onto these environments over time. Work in this research area includes the application of neural networks, genetic algorithms and case-based reasoning for learning user profiles (Jennings & Higuchi 1992; Baclace 1993; Sheth & Maes 1993; Hammond et al. 1994); and information-theoretic classification for categorizing documents based on users' preferences and on usage pattern analysis (Lang 1995; Armstrong et al. 1995).

These learning techniques are cast into single agent frameworks and do not, therefore, exploit the distributed and asynchronous flow of information in the aforementioned environments. In addition, the input space of an IF task is usually partially observable and non-stationary. Since these techniques follow a global, monolithic approach to the learning problem, their success in coping with the intricacies of the IF task may inherently be limited. Within the statistics and the computational learning communities, the idea of distributing a learning problem among a set of 'local experts' has been proposed for robustness against partial observability and non-stationarity (McLachlan & Basford 1988; Jordan & Jacobs 1994). These 'local experts' compete with each other in order to acquire local expertise in regions of the input space which may be overlapping. In the models so developed, gating of the experts is fixed and often depends on prior assumptions about the input data distributions. Furthermore, in these models the local experts compete so that a single winner typically accounts for each input example. In many learning tasks, however, it may be better that more than one expert accounts for a single example. This is the case for instance with the IF task where some articles occur independently of each other and others occur with dependencies (e.g. multi-threads, cross-postings, etc.). An additional advantage of a multiple expert approach to the IF task is that only a few experts may be applied combinatorially to account for a large set of example articles in the input space.

In this work, we therefore propose an adaptive model for multi-agent learning based on the metaphor of economic markets, that suits the characteristics of the IF task. The activities in an IF system are characterized by competition for resources, since users that arrive in the system are endowed with limited resources such as time and money, and different interests. The system contains agents which have asymmetric information and different inference technologies. Markets have been devised within economics for allocating limited or scarce resources among competing agents. They provide the machinery for decentralized decision-making where each agent processes only asymmetric and local information in order to evaluate decisions regarding goods and services. Because of their decentralized and local nature, markets can spontaneously be developed for the exchange of goods according to local needs; they can also adjust to unforeseeable changes.

Most of the market-like systems that are currently developed focus on finding an appropriate agent for solving a single task (Huberman & Hogg 1995). In addition, learning within these systems occurs either by mixing market and heuristic mechanisms for adjusting prices, or by auction mechanisms for determining market equilibrium prices at pre-specified users' preferences (Wellman 1994). For this reason, in the rest of this paper we develop various learning and adaptation techniques – i.e. reinforcement learning, bidding price adjustment and relevance feedback – and show how they are integrated into the multi-agent learning model of SIGMA (System for Information Gathering Market-based Agents).

2. Overview of SIGMA

Similarly to economic markets, the computational economy of SIGMA can be defined in terms of goods and agents that produce and consume those goods. The *goods* traded in SIGMA are information items (e.g. news articles) in different representation forms depending on the stage of processing. The agents are of two general categories: (i) the *consumer* agents and (ii) the *producer* agents. A consumer represents a user's goal and preferences for a filtering activity over a time period. The other category of agents, producers, transform goods from an input form into an output form according to their technologies. In response to a consumer's demand for goods they enter the local market and compete with each other to serve as efficiently as possible the demands for goods from other agents — consumers or other producers — within the market.

Figure 1 depicts the SIGMA computational market for netnews IF. There are two main types of producers[1]: (i) the Feature Extractors (FE) which transform each news article (a generic input good) into a document indexing representation; and (ii) the Profile Generators (PG) which are mid-producers since each of them takes as input a subset of the output of the FE producers and transforms it into a profile (a compact representation of documents) which the PG agent expects will satisfy the respective consumer's interests. The consumer in the SIGMA market is the Profile Selector (PS) agent. We next define the behaviors that are embedded into the FE, PG and PS agents.

FE Agent. Its purpose is to transform raw information contained in an article into a feature-based knowledge representation formalism that can be used to reason about the article. The vector space model (VSM) is used for implementing this formalism (Salton & McGill, 1983). The behaviors of an FE agent are: (i) request articles at regular intervals from the agent handling the Usenet repository and process them to create their VSM representation; and (ii) retrieve, on response to a PG agent's query specifying a set of newsgroups, all articles in VSM form which match the newsgroups in question. The inputs of an FE are raw articles containing structured information in their header part (author, location, newsgroup, etc.) as well as unstructured information in their text part. The outputs of an FE are the VSM representations of articles and

[1] In section 4 we introduce additional types of producers for managing resources and communication within SIGMA.

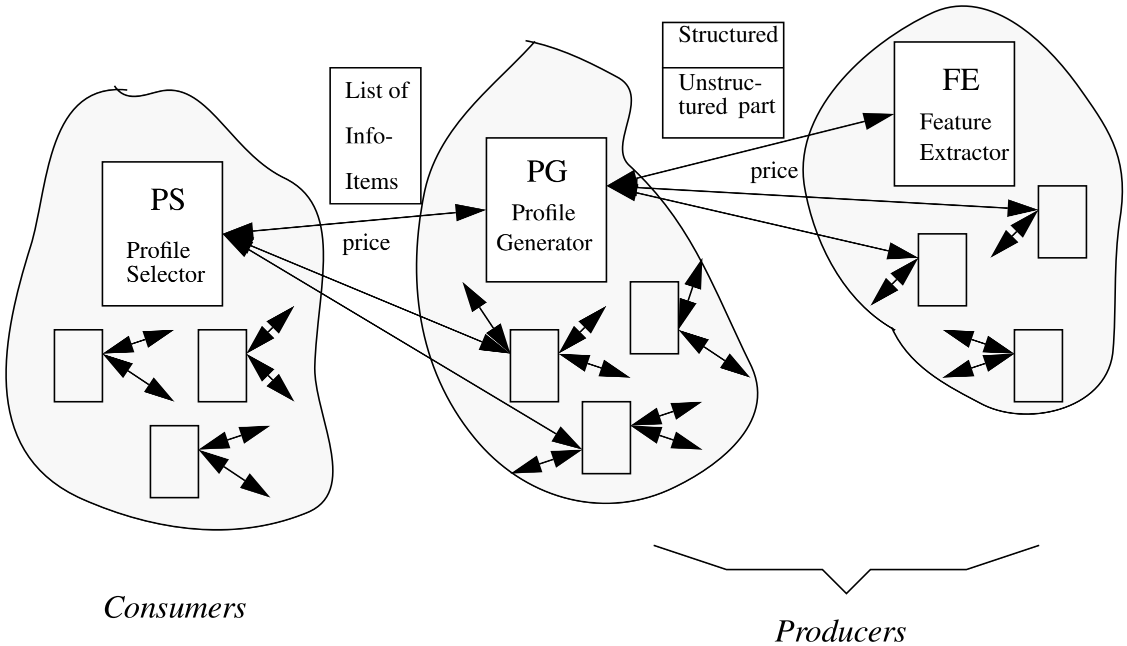

Figure 1: The flow of goods in the IF application of SIGMA.

word frequency tables. FE agents are distributed across newsgroups based on the first level of domain hierarchy (e.g. one FE for all 'comp.*' newsgroups).

PG Agent. Upon creation it is initialized with a profile that contains the same field values as the ones in the structured and unstructured parts of the query profile specified by the user. When selected at bidding the agent explores the newsgroups in its current profile and selects the articles that maximize its profits. The filtering technology that is employed for producing the selection of articles is based upon similarity metrics of the VSM representation. The behaviors of a PG agent are: (i) adjust the selling price of an article by using current performance data and submit it for bidding; (ii) retrieve VSM articles from appropriate FE agents, determine which articles to purchase (by employing profile similarity metrics calculated through the next behavior) such that profits are maximized and purchase those articles from the respective FE agents; (iii) maintain its profile using article evaluation feedback obtained through the PS agent and calculate profile similarity metrics. The inputs of a PG are VSM articles from FEs and article evaluation feedback from PS. The outputs of a PG are its bidding price and its collection of selected articles that are submitted to the PS agent.

PS Agent. It is endowed with a budget B which is a constraint on how much it can spend each time on buying articles from PGs at prices that depend on the bidding process and on the relevance of each article. Given this constraint the decision task of a PS agent at each time t amounts to buying from the set of the producers that have won the bidding at time t, a bundle of goods (i.e. articles) such that the user's cumulative rewards from buying goods in the long term is maximized. These rewards reflect the contribution of the articles towards satisfying the user's goal in the particular filtering activity. Through this feedback the PS agent should learn a policy that allocates its budget among the PS agents as satisfactorily as possible. The behaviors of a PS agent are: (i) construct a profile using the value and constraints of the user's query (ii) receive and select corresponding bids; (iii) select articles from PG agents such that utility estimates are maximized; (iv) update utility estimates with user feedback. The inputs of a PS agent are the bidding prices and the collections of articles from PG agents as well as the article evaluation feedback from the user. The outputs of a PG agent are the set of producers that win the bid and the set of articles that are selected for presentation to the user.

It is worth noting that the above three types of agents exhibit two levels of learning and adaptation within SIGMA. At the individual level the FE agents learn how to compress the noisy and partially structured information of an article into a set of keywords which should be significant for any filtering activity on that article. The PG agents learn, through user feedback, user profiles which represent local interests within possibly overlapping partitions of the input space. The PS agents employ user feedback to adapt their actions for selecting articles to ones that better satisfy the user's goal in a particular IF task. At the inter-agent level the PG agents compete with each other via the bidding mechanism and adapt their selling prices according to the PS's demand for their articles. The PS agents learn through user feedback how to combine the local profiles of the PG agents and map them into selections of articles that are as relevant as possible to the user's interests. Since a PS agent purchases articles from more than one PG agent this should enhance the robustness of the system with respect to the uncertainties of the input space. In the next section we develop the techniques which are incorporated into SIGMA for realizing the aforementioned learning and adaptation behaviors.

3. Learning and Adaptation in SIGMA

The type of good which is exchanged most and which is at the center of all learning and adaptation behaviors within SIGMA, is the VSM representation of an article. Like any other type of document an article contains structured information in its header part (author, subject, newsgroup, etc.) as well as unstructured information in its text part. A query submitted by a user to SIGMA consists of fields that correspond to the structured part of a news article — namely author(s), subject(s), newsgroup(s) and organization(s) — and a field that corresponds to the unstructured part — namely text keyword(s). We next present the four learning and adaptation behaviors embedded in SIGMA agents: (i) feature extraction in FEs, (ii) relevance feedback in PGs, (iii) bidding price adjustment in PGs and (iv) learning by mixtures of producers in PSs.

3.1 Feature extraction in FEs

The text of an article is first processed to eliminate noisy words by using a stop-list. It is then parsed by using Wordnet for extracting stemmed terms, namely nouns, adjectives, gerunds and past participles. These terms are used for calculating the term frequency $tf_{t,d}$ of term t in document d and for calculating the document frequency df_t of term t in all N documents that the particular FE agent has currently come across. In effect, each FE agent maintains a term frequency table using articles from the newsgroups to which the agent is allocated. The above frequency measures are used to calculate a weight for each term via the *tf-idf* formula (Salton & McGill, 1983), i.e.

$$w(t, d)) = tf_{t,d} \times \log(|N| / df_t) \tag{1}$$

The intuition behind this weighting scheme is that terms with high *tf-idf* values are unlikely to appear in a document by chance. The terms of the article with the top 20 sorted weights[2] are then assigned to the keyword field of the VSM representation whereas the rest of the terms are discarded. Given this representation, two documents can be compared for similarity by calculating the cosine of the angle between

[2] In Salton and Buckley (1987) a restriction on the number of words had only a minor effect on the performance of a retrieval system.

the two respective vectors of weights. This similarity measure is used by the PGs agents for selecting articles.

3.2 Relevance feedback and pricing in PGs

As already mentioned, upon creation, a PG agent is initialized with the query profile. Throughout its operation the agent updates its profile using the user evaluation feedback of the article (an integer in the range [0,5]) as forwarded by the PS agent. Since an article may be produced by multiple PGs the PS distributes the evaluation feedback in equal proportions to the respective PGs. The term weights of the PG's profile $w(t, PG_i)$ are updated according to

$$w(t, PG_i) = w(t, PG_i) + a \cdot r \cdot w(t, d) \tag{2}$$

where a is the learning rate and r is the reward received from PS. In addition, new terms are added to the profile by a lower learning rate that restricts their importance with respect to existing terms. The weights of the newsgroups in the newsgroup field of the PG's profile are similarly updated. When submitting a request for articles to an FE the PG sends the current list of newsgroups with τ additional newsgroups randomly picked from the domain of FE. This feature makes the PG agents proactive and enables them to cope with new newsgroups.

The task of a PG agent is to compete with other PG agents via the price bidding mechanism. If the agent wins the bid it produces at the bidding price articles to maximize its profits. The state variables determining the agent's perception of its opportunities for profit maximization are an index of its own efficiency level ω, defined as

$$\omega_i = \frac{\text{\#articles accepted}}{\text{\#articles proposed}} \times \frac{\text{\#times won bidding}}{\text{\#times bidded}} \tag{3}$$

and a vector determining the efficiency levels of its competitors s. When a PG_i agent sells an article at price p_i, with similarity between the article and its profile y_{ik}, it receives from consumer PS revenue $p_i \cdot y_{ik}$. Suppose each producer has constant marginal costs mc for producing an article; let $\bar{y}_i$ be the average similarity of articles produced in the past. Profits are then defined by

$$\Pi(\omega_i, s_i, \bar{y}_i) = [p(\omega_i, s_i) \cdot \bar{y}_i - mc] \cdot \sigma(\omega_i | s_i, \bar{y}_i) \tag{4}$$

where $\sigma(\omega_i | s_i, \bar{y}_i)$ is the market share of PG's goods as a function of their characteristics $\bar{y}_i$. It is given by the probability that a consumer PS will choose a good with average quality $\bar{y}_i$. Omitting details due to lack of space we estimate this probability by

$$\sigma(\omega_i | s_i, \bar{y}_i) = \exp[\omega_i - (p_i \cdot \bar{y}_i)] \Big/ \left[1 + \sum_{j=1}^{M} \exp[\omega_j - (p_j \cdot \bar{y}_j)]\right] \tag{5}$$

Since PG_i chooses prices to maximize profits, first-order conditions of (4) give

$$p_i = \left(\frac{1}{1 - \sigma_i} + mc\right) \Big/ \bar{y}_i \tag{6}$$

If a PG_i agent wins the bid it produces articles that have similarity higher than a particular threshold. This threshold can be set statically, or updated during learning by the PG_i agent.

3.3 Learning by mixtures of producers

The PS consumer wants to learn a mapping from user's interests to articles. The state of the consumer (x_t, z_t) is defined by an observable part, vector x_t whose elements are the values in the fields of the user's profile; and an unobservable part, vector z_t. Although some elements of z_t may be revealed through the learning process and assigned values, there will always be a part of the state that will remain unobserved. In addition, the processes that generate the state (x_t, z_t) as well as articles over Usenet are non-stationary. The task environment is therefore partially observable and non-stationary. To cope with such features the consumer learns the aforementioned mapping by learning how to fuse the inferences made by producers.

Thus, the task of the PS agent is to learn a policy that decides from which producer to buy the next article. The above state representation does not support this learning task. Instead, we use (ω_t, n_{tj}) where ω_t is the vector of efficiency levels of producers and n_{tj} is the number of purchases from producer PG_j. The PS agent learns a policy $\pi : \Omega \rightarrow A$ that maps agents' efficiency levels into actions of buying next article from a particular producer. The optimization task can be formulated in terms of Q-value functions as

$$Q_{\pi^*}(\omega_t, n_t, a_t) = E\{R(\omega_t, n_t, a_t) + V_{\pi^*}(\omega_{t+1})\} \tag{7}$$

where π^* is the optimal policy and $R(\omega_t, n_t, a_t)$ is the total net reward from buying n_t articles from producers at time *t*, i.e.

$$R(\omega_t, n_t, a_t) = \sum_{j=1}^{n_t} [r(\omega_t, n_{tj}, a_{tj}) + B_t - (p_{tj} \cdot y_{tj})] \tag{8}$$

We have developed a learning algorithm for implementing this learning behavior of the PS agent based on earlier work in (Karakoulas, 1995).

4. Discussion

The SIGMA market model described above has been implemented as a collection of specialized CALVIN (Communicating Agents Living Vicariously in Networks) agents. The CALVIN agent framework is an open architecture for facilitating the development of highly concurrent, embedded agent-oriented applications (Ferguson & Davlouros 1995). As such, the framework provides application developers with a powerful set of agent programming tools including libraries of intra- and inter-agent protocols, sensory and effectory apparatus, internal agent behavior APIs, persistent storage management, and support for preemptive (anytime) scheduling of behaviors, among others. Figure 2 shows the agent-level description of the system. In addition to the three main types of agents – FE, PG and PS – the system includes the Usenet Wrapper (UW) agent for retrieving Usenet articles from the network; the Repository Managers RM_1 and RM_2 for maintaining repositories of articles in VSM form; and the Bid Manager (BM) for brokering the bidding process between PG and PS agents.

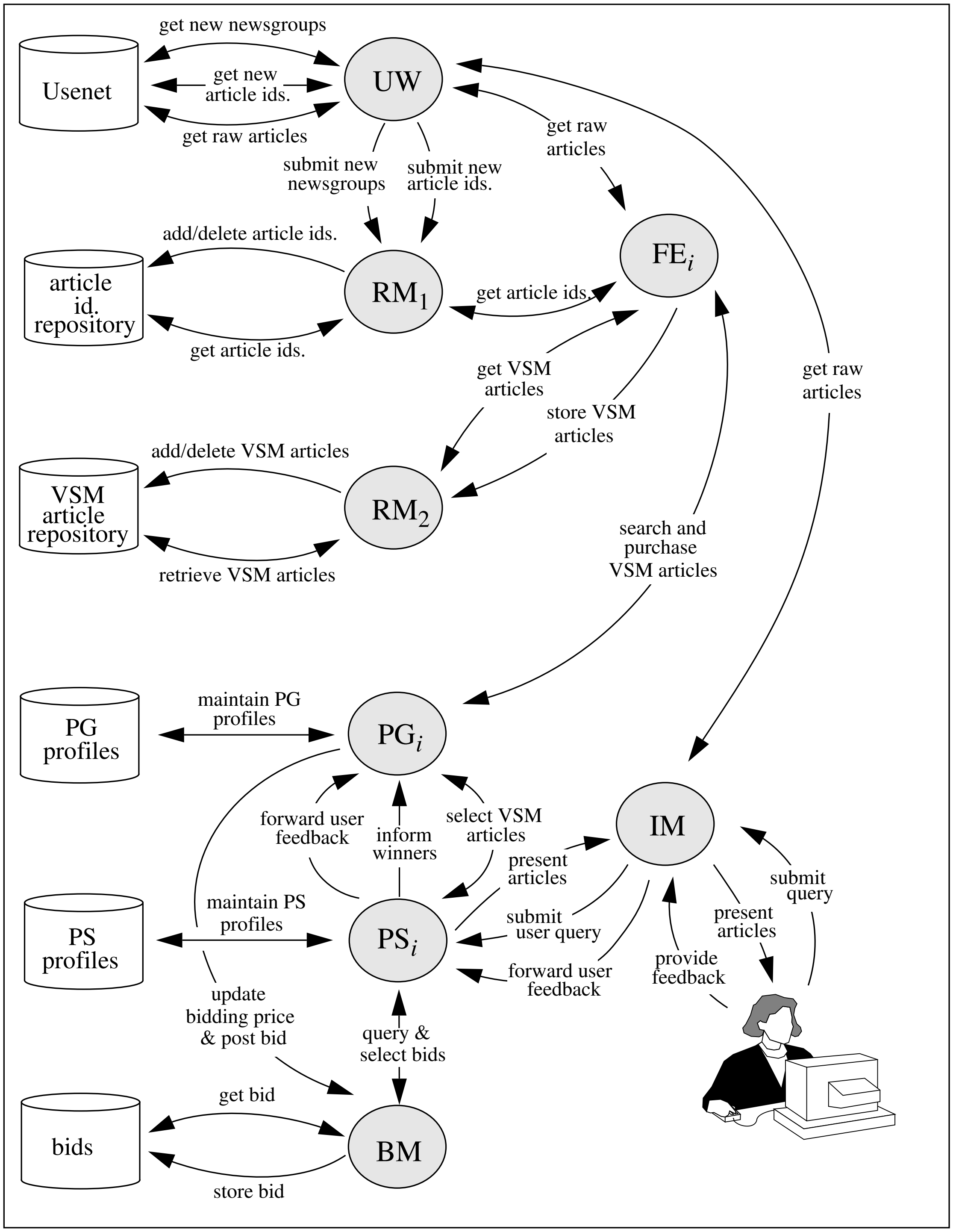

Figure 2: SIGMA agent-level system description.

The current focus of our work is on system evaluation. We plan to evaluate the IF application through two sets of experiments. The first of these concerns performance evaluation of the learning and adaptation techniques embedded in SIGMA. The main hypothesis of the proposed multi-agent learning approach is that it scales to the non-stationary and partially observable IF tasks better than monolithic learning approaches. We need to empirically evaluate this hypothesis by comparing the performance of SIGMA with one PG agent (i.e. a monopoly) with its performance when more than one PGs are employed; also how performance changes when the number of PGs increases. The adaptivity of the system will be evaluated by comparing SIGMA with a non-adaptive version of the system that does not include the PG and PS agents. The system should also be compared against other Usenet filtering systems that are available over Internet. The same user profile will be submitted to the systems under evaluation and their performance will be tracked on a daily basis. The second set of experiments involves measuring the potentially improved access to Usenet that is provided to users through SIGMA. For this purpose we are creating data collections of Usenet usage by recording subjects' actions while browsing their newsgroups on a daily basis over a period of three months. The respective volume of the original Usenet articles during this period is also stored. We plan to train SIGMA using the data of the first two months and then evaluate its performance on the data of the third month. Such evaluation should show how the system compares with the manual filtering of Usenet by the subjects.

References

Armstrong, R.; Freitag, D.; Joachims, T.; and Mitchell, T. 1995. WebWatcher: A learning apprentice for the world wide web. In *Proceedings of AAAI Spring Symposium on Information Gathering from Heterogeneous Distributed Environments*, Stanford, March 1995. AAAI Press.

Baclace, P.E. 1993. Personal information intake filtering. In *Proceedings of the Bellcore Information Filtering Workshop*.

Ferguson, I.A.; and Davlouros, J.D. 1995. On establishing multi-sensory multi-channel communications among networked users. In *Proceedings IJCAI Workshop on AI in Distributed Information Networks*, 103–109.

Hammond, K.; Burke, R.; and Schmitt, K. 1994. A case-based approach to knowledge navigation. In *Proceedings of AAAI Workshop on Indexing and Reuse in Multimedia Systems*, 46-57. American Association for Artificial Intelligence.

Jennings, A.; and Higuchi, H., 1992. A personal news service based on a user model neural network. *IEICE Transactions on Information Systems*, 75(2):198-209.

Huberman, B.A.; and Hogg, T. 1995. Distributed computation as an economic system. *Journal of Economic Perspectives*, 9(1):141-152.

Jordan, M.; and Jacobs, R. 1994. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6, 181-214.

Karakoulas, G.J. 1995. A Q-Learning approach to cost-effective classification. Technical Report NRC-38390, Knowledge Systems Laboratory, National Research Council, Ottawa, ON, Canada.

Lang, K. 1995. NewsWeeder: Learning to Filter Netnews. In *Proceedings of the Thirteenth International Conference on Machine Learning*, Tahoe City, CA, 331-339. Morgan Kaufmann.

McLachlan, G.; and Basford, K. 1988. *Mixture models*. Marcel Dekker.

Salton, G; and McGill, M. 1983. *Introduction to modern information retrieval*. McGraw-Hill.

Salton, G.; and Buckley, C. 1987. Term weighting approaches in automated text retrieval. Technical Report 87-88, Cornell University, Department of Computer Science.

Sheth, B; and Maes, P. 1993. Evolving agents for personalized information filtering. In *Proceedings of the Ninth Conference on Artificial Intelligence for Applications*, 345-352. IEEE Computer Society.

Wellman, M.P. 1994. A computational market model for distributed configuration design. In *Proceedings Conference of the American Association for Artificial Intelligence*, Seattle, WA, 401–407.