Information Filtering via Hill Climbing, WordNet, and Index Patterns

Abstract

The recent explosion in Internet growth has left many users awash in a sea of information, and has spurred the need for intelligent filtering systems. This paper describes work implemented in the INFOS (Intelligent News Filtering Organizational System) filtering system that is designed to reduce the user's search burden by automatically categorizing data as relevant or irrelevant based upon user interests. These predictions are learned automatically based upon features taken from input data articles and collaborative features derived from other users. The actual filtering is performed via a hybrid technique that combines a keyword-based hill climbing method, the knowledge-based conceptual representation of WordNet, and partial parsing via index patterns. A hybrid system integrating all approaches combines the benefits of each while maintaining robustness and scalability. INFOS has been tested upon Usenet news articles and preliminary tests have been performed on WWW pages.

1. Previous Work

While some filtering systems attempt to summarize input articles or extract key data (Mauldin, 1990; Soderland & Lehnert, 1994), the goal of INFOS is simply to determine whether or not the user is interested, or not interested, in a particular article. For example, a news stream may contain many news articles and INFOS's job is to determine which articles out of the data stream the user wants to read. As a result, the reader saves valuable time that would otherwise be spent sorting through messages. The criteria for determining what a user finds interesting varies greatly from one individual to another, implying that the system must be capable of adapting to variable user interests.

One of the most popular methods implemented in news filtering is to extract *keywords* or *tokens* from the text to use as features for classification. Some systems "watch" for predefined keywords which match a rule to determine what action to take (Stevens, 1992) while many other systems use all words in the input article as features after passing through a stemmer or stop list (Eberts, 1991; Jennings & Higuchi, 1992).

Once keywords have been extracted, one of the most widely used methods for classification using these features is *tf-idf* (Salton, 1991). In tf-idf, the term-frequency is coupled with the inverse document frequency to provide a metric of relevancy between documents based upon the frequency of occurence of each extracted feature in the individual document and among all documents. By combining terms from a document and user model (or query) to form a vector, the document vector closest to the user model vector is retrieved as the best match, classifying the input document. Tf-idf is the classification method implemented in the NewT system (Sheth, 1994), which is based upon tf-idf and genetic algorithms. The tf-idf method is also compared against Lang's MDL method as a baseline for evaluation in NewsWeeder (Lang, 1995).

DeJong (1982) and Mauldin (1991) have explored knowledge-based approaches to news filtering that are based upon conceptual processes. The advantage of a symbolic, knowledge-based approach is that the input text is understood as a human might understand the text, allowing for much greater understanding (Ram, 1992). Both Dejong's FRUMP and Mauldin's FERRET systems are based upon Schank's Conceptual Dependency (CD) theory (Schank, 1977), which is intended to be an unambiguous representation for knowledge. The systems parse input articles into a CD representation and compare the information to *scripts*, or stereotypical sequences of events, to determine the content of the article.

In addition to filtering techniques based upon the direct features of an article, filtering may also be performed based upon indirect features. Social or collaborative filtering is a recent area of activity which follows the indirect path to classify news, mail, and WWW documents (Brewer & Johnson, 1994; Lashkari et. al., 1994; Mock & Vemuri, 1994; Resnick et. al., 1994; Goldberg et. al, 1992). In collaborative filtering, a population of users reads and classifies the input articles. These classifications are made publicly available as reviews. The reviews then become an additional input for the filter. As a result, other users may decide to read an article based upon the reaction of their peers.

2. Filtering via Global Hill Climbing

As a result of usability studies, an important requirement for filtering systems is that the user model is that it must be very simple for users to understand. Moreover, the interests of users change quickly, implying the need for user to be able to manually specify what should be filtered. If the model employed in the filtering system is too difficult to change or understand, the typical reader will never use it (Stevens, 1992). For example, a neural network model (Eberts, 1991) is almost impossible for a user to change. The tf-idf method is considerably simpler, but users may have difficulty understand how the term frequency and inverse document frequency terms inversely relate to each other. In psychological experiments, subjects preferred negation/addition strategies over reciprocal/inverse strategies (Collis, 1963). Consequently, the approach taken in INFOS has been to design a system which can be easily modified with a single control for weighting each term, but still support adequate filtering performance.

2.1 Global Hill Climbing Algorithm

The method implemented in INFOS is a linear discriminant method based on a table of features and is termed Global Hill Climbing (GHC). A summary of GHC is described below and details may be found in the literature (Mock, 1996). The table counts the number of times each feature has been found in each class. Since the table contains only one variable per class, it is simple for users to understand and manipulate. The table is created via hill climbing with intrusively collected data. As the user reads messages, she is prompted by INFOS to indicate if each message read was accepted (liked) or rejected (disliked). The outcomes are used to increment the table's weights accordingly.

An example of a GHC table is shown in Table 1. The word "agent" has been extracted from eight articles as a feature, and all eight times it has been in accepted articles. Similarly, the author feature of “mock@intel.com” has appeared in three accepted articles and three rejected articles. This data indicates that the author “mock” may not be particularly predictive, but articles containing “agent” are likely to be of interest. Conversely, articles containing “flames’ are likely to be of disinterest. In addition to using words from the articles as features, other types of features such as collaborative data may also be included. These other users are local users running the same news system who are willing to share their own reviews with others. In Table 1, user “Crisha” has accepted six articles the current reader has accepted, and she has rejected one article the current user has accepted. Similarly, Crisha has rejected two articles the current user has accepted, and rejected three articles the current user has rejected. This table indicates that the current reader’s accepted messages strongly correspond with Crisha’s accepted messages, while the current reader’s rejected messages may not correspond with Crisha’s rejected messages. As new articles are read, the table grows and more closely matches user interests automatically.

Word	**Accepted**	**Rejected**
agent	8	0
flames	2	7
mock@intel.com	3	3
Crisha Accepted	6	1
Crisha Rejected	2	3

Table 1: Global Hill Climbing Table of Weights

Once the GHC table is constructed, classification of new messages is performed by extracting the features from the new article and then computing the sum of all the Accepted and Rejected values from matching features in the table. If the Accepted percentage subtracted from the Rejected percentage exceeds *A*, the message is classified as being of interest. Conversely, if the Rejected percentage subtracted from the Accepted Percentage exceeds *A*, the message is classified as being of no interest. Messages in between are marked unknown. *A* was set to 0.15 so that some margin of difference was necessary to classify a message, but this value has been left as a user-adjustable setting. The classification process for a set of feature terms *t* is expressed mathematically by:

$$SimilarityPercentage(class)_t = \frac{\sum_t ClassOccurrences_t}{\sum_t TotalOccurrences_t} \tag{1}$$

$$Class_t = \begin{Bmatrix} \left(SimilarityPerc(Acc)_t - SimilarityPerc(\mathrm{Re}\,j)_t\right) > A : Accepted \\ \left(SimilarityPerc(\mathrm{Re}\,j)_t - SimilarityPerc(Acc)_t\right) > A : \mathrm{Re}\,jected \\ else : Unknown \end{Bmatrix} \tag{2}$$

If only equation 1 and 2 were used for classification, authors, text from the body, text from the subject, and collaborative features will all be counted and combined in the same way. A result of this method is a bias toward those features that occur most often, a result that may not be desired. For example, the word “agent” may appear many times in the body of an article, but the author will only appear once. However, the “agent” feature is not necessarily more important than the author simply because the structure of a news article dictates that it appear more often.

A solution to this problem is to separate the global hill climbing table into a set of individual tables with one table for each type of feature. Percentages of acceptance and rejection can be computed from the features among each table, and then these percentages combined to compute the final classification:

$$SimilarityCombn(Class)_t = \begin{array}{l} K_1 \times SimilarityPerc(Class)_{author} + K_2 \times SimilarityPerc(Class)_{sub} + \\ K_3 \times SimilarityPerc(Class)_{text} + K_4 \times SimilarityPerc(Class)_{collaborative} \end{array}$$

$$Class_t = \left\{ \begin{array}{c} \left(SimilarityCombn(Acc)_t - SimilarityCombn(\mathrm{Re}j)_t\right) > A: Accepted \\ \left(SimilarityCombn(\mathrm{Re}j)_t - SimilarityCombn(Acc)_t\right) > A: \mathrm{Re}jected \\ else: Unknown \end{array} \right\} \quad (3)$$

What values should be assigned to constants K_1 through K_4? Some systems (Jennings & Higuchi, 1992) give higher weight to the subject features on the assumption that these are most predictive. To investigate which terms are actually most predictive, experiments were performed to evaluate the impact of each feature individually. The features were then combined based upon how much impact they showed individually; i.e., the most predictive feature was given the highest weight, and the least predictive features given the lowest weights.

The data used for the experiments consisted of news articles randomly selected from the **ucd.life** newsgroup. This newsgroup was selected since it covers a variety of general topics presumably of interest to the test subjects, all UCD students. When processing articles, words from each article were passed through a stop list. Binary encoded files were also thrown out, and quoted text from old articles removed. A total of 14 users read 100 sequentially posted messages and marked each as accepted, rejected, or unknown. From these 100 messages, 50 messages were randomly selected for training, and the system predicted the users’ choices for the rest of the messages using equation 2. These predictions were one of three classes: Suggested, Not Suggested, or Unknown. The predictions were then compared to the actual classifications provided by the subjects.

The results were summarized by Mock (1994) and indicate that classification via the subject features results in the highest percentage correct (52%) with the lowest error (12%), probably since subject words are accurate predictors of entire threads that may be of interest. The textbody features actually give the largest percentage correct (54%), but also give the largest error (19%). Collaborative filtering gave the next best results (46%), and author alone was the worst predictor (38%), although not far behind the others. All methods perform better than chance. Based upon these results, a value of 0.35 was assigned to K_2, the subject’s weight, 0.25 to K_3 and K_4, the collaborative and textbody weights, and 0.15 to K_1, the author’s weight. Using these weights and equation 3, the classification process was rerun and the results shown below.

Percentage Correct Classifications	51.5%
Percentage Incorrect Classifications	7.3%
Percentage Unknown Classifications	40.9%
Within Error, Percent of False Positives:	50%
Within Error, Percent of False Negatives:	50%

While the combined classification scheme results in a slightly lower correct classification rate than the subject-alone method, the error rate is much lower than the subject-alone error rate.

3. Case-Based Filtering via WordNet

While GHC is simple and produces good results, it is unable to discern fine differences in features since it linearly combines all input features through the conditional independence assumption; e.g., if we are not interested in messages with the features "king" and "queen" but we are interested in messages with the features "Sacramento" and "king" (the basketball team), then the global method will be using the same accepted and rejected values for the word "king" and may be unable to classify correctly these articles. The underlying problem is that the system does not understand the actual concept behind the words or phrases.

3.1 WordNet

To address these problems in INFOS, the WordNet knowledge base hierarchy is combined into a case-based reasoning system. By retrieving individual cases and using the classification of those cases to classify new articles, the system is capable of avoiding the limitations of linearity. By combining these knowledge-based techniques with the GHC techniques previously described, the system is capable of growing through the keyword feature tables while also retaining semantic knowledge through WordNet. Additionally, a CBR system also provides an excellent opportunity to support information retrieval of previously read articles in addition to information filtering.

WordNet is an on-line lexical reference system inspired by current psycholinguistic theories of human lexical memory that organizes nouns, verbs, adjectives and adverbs into synonym sets (Miller, 1995). Different relations link the synonym sets. INFOS makes use of the hierarchical organization of synonym sets. With approximately 107,000 noun senses and 27,000 verb senses, the current version of WordNet (v1.5) is about the size of a paperback dictionary.

An example of the WordNet ISA hypernym hierarchy for the word "hacker" is shown in figure 3. When a word is found in the WordNet lexicon, all definitions or *senses* of that word are provided. The figure displays two noun definitions; one for the golfer, and the other for a programmer. These definitions are organized hierarchically, from the most specific up to more abstract concepts along ISA links.

```
Sense 1
hacker
       => golfer, golf player, linksman
```

```
        => player, participant
           => contestant
              => person, individual, someone, mortal, human, soul
                 => life form, organism, being, living thing
                    => entity
                 => causal agent, cause, causal agency
                    => entity

Sense 2
hacker
     => programmer, computer programmer, software engineer
        => engineer, applied scientist, technologist
           => person, individual, someone, mortal, human, soul
              => life form, organism, being, living thing
                 => entity
                => causal agent, cause, causal agency
                  => entity
```

Figure 3 : Sample WordNet hypernym hierarchies for the word "hacker."
Two sense definitions, organized from the specific to the general, are displayed.

If INFOS indexed news articles based upon all the sense definitions of nouns and verbs found in an article, then a large number of irrelevant indices would be created due to multiple word meanings. Consequently, INFOS attempts to find appropriate noun or verb phrases based upon Paice's index extraction algorithm (Paice, 1989). filtering. This algorithm assumes that sentences repeat an underlying concept within a "topic neighborhood" of a few sentences. Those words occurring with a high frequency are likely to be relevant to the topic at hand.

Paice's algorithm was modified to operate upon WordNet word sense definitions rather than individual words. First, verbs and nouns from each sentence are identified through WordNet and their hierarchical definition referenced. This step results in a linked list of nodes, where each node contains the hypernym sense definition for the nouns and verbs in that sentence. Since each word is expanded into all possible sense definitions of that word, this pool of sense definitions may not accurately reflect the actual topic. For example, in the sentence "the ocean is cold.", both definitions of ocean from figure 3 will be expanded into the node list. However, only the body of water definition is relevant; the large-quantity definition does not apply. To refine the sense definitions and select relevant ones, neighborhoods of sentences are examined and the intersection of sense definitions that match within a specified neighborhood are selected. This process restricts the selected definitions only to those that are reoccurring topic stems and are then more likely to be relevant to the document. Only the first 20 sentences of articles were processed to speed execution in the event of extremely long postings. The assumption was also made that long messages will contain relevant material at the beginning of the article. Details of the algorithm may be found in Paice's work (1989).

After candidate nouns and verbs have been identified, this information is used to index the document. In addition to the sense definition itself as an index, other relevancy statistics are also associated to each term, including frequency and rarity (Evans et. al, 1991). *Frequency* is merely the number of times the term appears in the document / number of times the term has appeared in the domain. This measure operates upon the assumption that domain words appearing often are indicative of the document. *Rarity* is a measure of the expected frequency of a word in general English. In WordNet, this is obtainable through a terms polysemy count. A common word such as “system” has a high value of 15, while a rare English word such as “cilia” has a low value of 2. In INFOS, the rarity R is defined as:

$$R_{sense} = 1 - \left(\frac{Polysemy_{sense}}{Max\_Polysemy} \right)$$

Based on this definition, extremely rare words will have a value of 1, while common words will have a value closer to 0. Extremely common words will likely be filtered out by the stop list.

Once both frequency and rarity have been determined, the two are multiplied together to give a general relevancy statistic for a sense term:

Relevancy $R = Rarity \times Frequency$ for each term. (4)

The relevancy value is stored with each term and is used in memory retrieval to determine how closely an old article matches a new document.

5.2 Indexing of Cases

Once the appropriate noun and verb phrase senses have been extracted from a textual case, the article is saved and the senses used to index the case. When concepts cannot be extracted (e.g., with novel words or domain-specific slang words and expressions) these terms are used directly to index the case through an inverted index. In this way, cases are indexed via both conceptual (controlled) and keyword (uncontrolled) vocabularies, allowing conceptual retrieval when possible, and also keyword retrieval under unforeseen situations so that performance is still possible (Callan & Croft, 1993).

The method in which articles are indexed using the sense definitions is to construct a pointer to the file that contains currently defined sense in a global abstraction hierarchy. An example memory hierarchy with three cases is shown in figure 4. In this example, one article contains the word “vehicle,” another article contains the word “bicycle,” and the last article contains the word “car.”

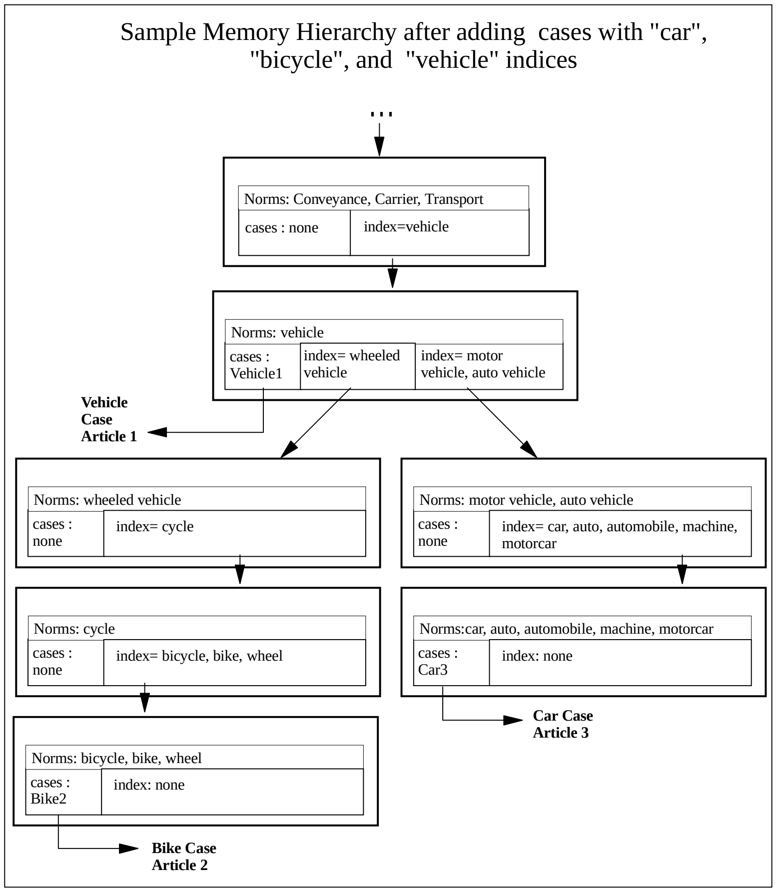

Figure 4: Sample Memory Hierarchy for Indexing Cases

In figure 4, the root node is not shown, but the sub-hierarchy starting at the Conveyance concept is displayed. This node represents the concept regarding items of transport and conveyance. All sub-nodes inherit the norms of their ancestors, hence all nodes located below this must also refer to transportation vehicles. One specific index currently exists from the Conveyance node, and it points to the Vehicle node. In turn, the Vehicle node points to sub-nodes, one regarding cycles and another regarding automobiles. In addition to pointing to sub-nodes, the Vehicle node also has an index to a specific case (news article) referencing vehicles. In a similar fashion, indices from the wheeled vehicle and the auto nodes are further specialized until they too point to actual cases referencing those concepts.

5.3 Memory Retrieval

Case-based memory retrieval involves searching for applicable cases based upon a given set of features. These features are simply WordNet sense indices from a new article that needs to be classified. Case retrieval in INFOS is fairly simple. A depth-first search is performed in the memory abstraction hierarchy along indices that match the input query until cases are retrieved or there are no more input indices to follow. To allow for partial matches (e.g., retrieve cases regarding bicycles when the input is about cars), path mismatches in the hierarchy can be traversed until an error threshold value is exceeded. In INFOS, 85% of the concepts had to match for a case to be retrieved.

For each case that is retrieved, the overall match value for that case is computed by summing over all *n* feature queries the following distance function:

$$Match = \frac{1}{n}\sum_{i=1}^{n}(MatchPercent_i) \times \mathrm{Re}levancy_i \qquad (5)$$

In INFOS, the retrieved cases are sorted by degree of match. The classification statistics of the best matching case can then be used to classify the new article, using the Accepted and Rejected counters for the case and computing a classification via equation 2. The article number can also be displayed as a justification to the user to indicate why INFOS believes new articles should be classified similarly.

5.4 Results of Case-Based Scheme

The same tests and testing methodology that was used to evaluate the global hill climbing scheme was also run with the case-based scheme. Finally, the case-based scheme was tested when used in conjunction with the global scheme. In this mode, the global scheme classification was performed first. If the global scheme returned an unknown classification, then the classification of the case-based scheme was used. The global scheme was performed first because it was found to have a lower error rate, and is also quicker than the CBR method. If a simple method can provide the correct classification, it is reasonable to use that method before more complex ones are attempted. The overall results indicate that the semantic and keyword based filtering provided by the combined best match CBR and global hill climbing scheme performs best. The hybrid method improved the correct classification percentage over the global hill climbing method alone.

A summary of the results is shown in Table 3 depicting the Global Hill Climbing method (GHC), CBR alone using the best match to classify of the most highly ranked case as the classification, and the CBR scheme combined with the global hill climbing method. The combined CBR scheme using best match with global hill climbing performed best, although it had a slightly higher error rate than the global hill climbing method alone.

Classification Method	Percentage Classified Correctly	Percentage Classified Unknown	Percentage Classified Incorrectly	Percent False Positives	Percent False Negatives
Global Hill Climbing (GHC)	51.5	40.9	7.3	50	50
CBR - Best Match as Class	39.8	50.5	9.5	77	33
Combine GHC + CBR Best Match	58.0	29.9	12.1	62	37

Table 3: Classification accuracy for Hill Climbing, CBR, and Hybrid Methods

The results from this experiment indicate that the global hill climbing method still has the lowest error but the combined scheme provides the best correct classification rate. The case-based scheme will have some poor indices due to the sense disambiguation problem that can allow irrelevant cases to be retrieved. Consequently, the CBR method has a higher error rate than the global hill climbing method. When combined with the global hill climbing scheme, the best match CBR method does achieve a higher correct classification percentage at 58%, although it suffers from a slightly higher error rate of 12%.

6. Future and Ongoing Work

In addition to the hybrid methods of global hill climbing and case-based reasoning via WordNet, ongoing work with INFOS also incorporates genetic algorithms to explore the news space, and index patterns or scripts to parse input articles more accurately. Other areas of proposed work include modifications for INFOS to run offline, a graphical user interface, self-modifying parameters, the incorporation of other knowledge bases such as CYC (Lenat, 1995), and the application of filtering to the World Wide Web and intelligent tutoring systems. Additional information is available from the WWW at:

http://phobos.cs.ucdavis.edu:8001/~mock/INFOS/infos.html

7. Summary

As the information age grows in scale, the amount of incoming data becomes too large for humans to handle. The internet has been growing a tremendous rate. Gigabytes of news articles flow through the internet daily, and World Wide Web pages number around 10 million. The central issue in this work addresses methods to model user interests automatically so that this data, usenet news articles in particular, can be filtered intelligently. However, in order to be a useful tool, the user model must be capable of adapting to user interests, articles must be displayed to give as much information as possible so users can intelligently browse and select articles to read, users must be capable of modifying and understanding the user model constructed for them, and the news filtering system must give accurate predictions.

Knowledge-based systems have the advantage of analyzing input text in detail, but at the cost of computational complexity and the difficulty of scaling up to many domains or domains of large scale. In contrast, statistical and keyword approaches scale up readily but are limited to a shallower understanding of the input. A hybrid system implemented in INFOS that integrated a keyword hill climbing method with a case-based reasoning method improved classification over the keyword method and provided scalability along with domain knowledge. However, the case-based approach did introduce some additional error due to the lack of robust disambiguation.

8. References

Brewer, R.S. & Johnson, P.M. (1994). Toward Collaborative Knowledge Management within Large, Dynamically Structured Information Systems. *Internal Research Report, Collaborative Software Development Laboratory, Department of Information and Computer Sciences, University of Hawaii*. WWW: http://www.ics.hawaii.edu/~csdl/urn.

Callan, J.P. Croft, W.B. (1993). An Approach to Incorporating CBR Concepts in IR Systems. *Proceedings of the 1993 Spring Symposium on Case-Based Reasoning and Information Retrieval*, AAAI Press, pp 28-32.

Collis, K.F. (1980). Levels of Cognitive Functioning and Selected Curriculum Areas. In J. Kirby and J. Biggs (Eds.), Cognition, Develoment, and Instruction. New York, NY: Academic Press, pp. 65-89.

DeJong, G. (1982). An Overview of the FRUMP System. In W.G. Lehnert & M. H. Ringle (Eds.), Strategies for Natural Language Processing, Hillsdale, NJ: Lawrence Erlbaum, pp. 149-174.

Eberts, R. (1991). Knowledge Acquisition Using Neural Networks for Intelligent Interface Design. *Proceedings of the 1991 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1331-1335.

Evans, D.A. Ginther-Webster, K. Hart, M. Lefferts, R.G. Monarch, I.A. (1991). Automatic Indexing Using Selective NLP and First-Order Thesauri. *Proceedings of the Intelligent Text and Image Handling Conference*, Barcelona, Spain. pp 624-643.

Goldberg, D., Nichols, D., Oki, B., Terry, D. (1992). Using Collaborative Filtering to Weave an Information Tapestry. *Communications of the ACM*, **35** (12), pp. 61-70.

Jennings, A. & Higuchi, H. (1992). A Personal News Service Based on a User Model Neural Network. *IEICE Transactions Inf. & Systems*, **E75** D(2), pp. 198-209.

Lang, K. (1995). NewsWeeder: Learning to Filter Netnews. *Proceedings of the Twelfth International Machine Learning Conference.*

Lashkari, Y., Metral, M., & Maes, P. (1994). Collaborative Interface Agents. *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pp. 444-449.

Lenat, D.B. (1995). CYC: A Large-Scale Investment in Knowledge Infrastructure. *Communications of the ACM*, **38** (11), pp. 32-38.

Mauldin, M. L. (1991). Conceptual Information Retrieval: A case study in Adaptive Partial Parsing. Kluwer Academic Publishers. Norwell, MA.

Miller, G.A. (1995). WordNet: A Lexical Database for English. *Communications of the ACM*, **38** (11), pp. 39-41.

Mock, K. (1996). AAAI paper.

Mock, K. (1996). Ph.D. Dissertation.

Mock, K., Vemuri, V. (1994). Adaptive User Interface for Intelligent Information Filtering. *Proceedings of the Third Golden West International Conference on Intelligent Systems*, pp 506-517.

Paice, C.D. (1989). Automatic Generation and Evaluation of Back-of Book Indexes. *Prospects for Intelligent Retrieval, Informatics 10*, Cambridge MA.

Ram, A. (1992). Natural Language Understanding for Information-Filtering Systems. *Communications of the ACM*, **35** (12), pp. 80-81.

Resnick, P., et al. (1994). GroupLens: An Open Architecture for Collaborative Filtering of Netnews. *Internal Research Report, MIT Center for Coordination Science.*

Riecken, D. (1994). Intelligent Agents. *Communications of the ACM*, **37** (7), pp. 18-21.

Salton, G. The SMART Retrieval System: Experiments in automatic document processing. Englewood Cliffs, NJ: Prentice-Hall, 1971.

Schank, R.C. & Abelson, R. (1977). Scripts, Plans, Goals, and Understanding. Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Sheth, B.D. (1994). A Learning Approach to Personalized Information Filtering. Masters Thesis. Department of Computer Science and Engineering, Massachusetts Institute of Technology.

Soderland, S. and Lehner, W. (1994). "Corpus-Driven Knowledge Acquisition for Discource Analysis." Proceedings of the Twelfth National Conference on Artificial Intelligence, pp. 827-832.

Stevens, C. (1992). Knowledge-Based Assistance for Accessing Large, Poorly Structured Information Spaces. Ph.D. Dissertation Thesis. Department of Computer Science, University of Colorado.