# DOCUMENT FILTERING: INTELLIGENT INFERENCE SYSTEM FOR WEB

**Miss.Rasika G.Charate***

**Dr.P.N.Chatur****

**Prof.K.P.Wagh*****

## ABSTRACT

I develop a new case-based approach for text document filtering based on automatic construction of filtering profiles using Bayesian inference network learning for web. Bayesian inference networks, based on probability theory, offer a suitable framework to harness the uncertainty found in the nature of the filtering problem. In order to learn the networks effectively, Explore three different techniques for Discretization. Good features of high predictive power are automatically obtained from the training document content. This approach does not need to know in advance the subject or content of documents as well as the information needs expressed as topics. The system is capable of selecting HTML/text documents, collected from the Web, according to the interests and characteristics of the user. A series of experiments on a set of topics were conducted on two large-scale real-world document corpora. The empirical results demonstrate that our Bayesian inference network learning with advanced Discretization achieves

* Department of Computer Engineering, Government College of Engineering , Amravati, Maharastra, India

** Head of Department, Department of Computer Engineering, Government College of Engineering , Amravati, Maharastra, India

*** Department of Information Technology, Government College of Engineering , Amravati, Maharastra, India

better performance over the simple Naive Bayesian approach. Presently the system acts as an intelligent interface for the Web search engines

**KEYWORDS**

Information filtering, User modeling, World Wide Web, user profiles, Case-Based Reasoner

## INTRODUCTION

The Internet has rapidly become the main route for information exchange world-wide. Besides the problem of bandwith, the growth of Internet and the World Wide Web makes it necessary for the end user to cope with the huge amount of information available on the net. Filtering information is a problem becoming increasingly relevant in information society. The issue of information filtering involves various kinds of problems, such as (i) designing efficient and effective criteria for filtering, and (ii) designing a friendly, non-obtrusive, intelligent interface to lead the user to the most interesting information, according to her/his interests. In this work I present an Information Filtering system, have developed for selecting HTML/Text documents from the World Wide Web. The system selects the documents according to the interests (and non-interests) of the user, as desumed by the system through the interaction. To do so, the system makes use of a User Modeling ad-hoc subsystem, particularly conceived for Internet users. One distinguishing feature of the presented system is its hybrid architecture: a combination of a Case-Based Reasoner with a sub-symbolic module (here, an artificial neural network). The evaluation of the system is based on an empirical approach and makes use of a non-parametric statistics for testing hypotheses on the system behavior.

## 1.RELATED WORK

Information Fitering become most important system in day to day life for users.from since years many Information systems has been developed using Probabilities and other approaches. I analyze their limitations and present a motivation of proposed approach.

### 1.1 Related Work Using Probabilistic Approaches

Fuhr and Pfeifer [7] investigated probabilistic information retrieval methods. Using abstraction concepts, this approach combines classical retrieval models with logistic regression. It mainly focuses on automatic indexing from document titles and abstracts, with index terms drawn from a dictionary of descriptors. The dictionary contains term-descriptor rules, term-specific, and descriptor-specific information. The indexing process identifies terms in form of occurrence.

Tzeras and Hartmann [20] proposed a similar approach for indexing terms from a prescribed dictionary with mapping rules for terms and descriptors. They proposed a Bayesian inference network approach for developing the model. A common assumption of these two approaches is the existence of a pre-defined indexing dictionary. Moreover, they concentrate on developing the association between the indexing descriptors and the document content.

Fung and Del Favero [24] applied Bayesian inference networks to filtering. It allows users to specify the set of topics of interests and the system is able to filter relevant ones from incoming time-sensitive documents. A key characteristic of this approach is that semantic relationships between topics can be specified.The relationships are used for improving the retrieval effectiveness by constructing a multiple-topic Bayesian inference network.One assumption of this approach is the existence of topic description. Scalability is another concern since the topology of the network will increasingly complicated even if a moderate number of topics involve.

InRoute was developed by Callan [3]. It is an IF system for filtering text documents. It requires the user to specify the information need in the form of either query language or natural language. Both the topic profile and the document profile are represented as inference networks. The topic profile is constructed based on the natural language or the query language supplied by the user. The system then changes the representation of the network. There is no mechanism for the system to process feedback

data collected from the users. A profile will not change unless the user change it manually or build a new one.

Pazzani and Billsus employed a simple Bayesian classifier technique to learn a user profile [15]. They applied this technique for identifying interesting Web sites based on the relevance feedback from users.

### 1.2 Related Work Using Other Approaches

NewsWeeder, developed by Lang [14], is an IF system for filtering Usenet news. It learns the filtering profile from documents rated by users. Currently, there are six ratings, namely, essential, interesting, borderline, boring, gong and skip. These ratings are used by the system as feedback data to learn the profile. The techniques used by NewsWeeder are based on vector representation of documents and the Minimum Description Length (MDL) method. The documents processed by the system are represented by using vectors composed of tokens which can be words or a combination of words or punctuation. A probability distribution of the tokens is calculated for each rating. The MDL measure is used to find the best distribution. The MDL principle provides an information-theoretic framework for balancing the tradeoff between model complexity and training error.

NewT is an IF system developed by Sheth and Maes [18], [19] and it is used for filtering Usenet news. Like NewsWeeder, it can also learn the profiles from documents rated by users. After reading news, the user gives some ratings and the system will process the rated documents and construct a profile. It uses a vector representation for documents and it employs a genetic algorithm to discover new profiles. It also has a user friendly graphic user interface. One disadvantage of NewT is that it uses a keyword-based approach for searching documents and some concepts may not be able to be represented by just several keywords.

SIFT is an IF system for filtering Usenet news developed by Yan and Garcia-Moline [23]. The user can access the system from both the WWW interface or by sending and receiving emails. To use the system, users need to build their own profiles. Users can test the profiles before setting them for operation. The profiles are represented in twoways, namely, the Boolean model and the vector space model. Requiring manual construction and change of the profiles is a disadvantage for SIFT. Automatic learning of a profile can alleviate this problem. Furthermore, SIFT does not have a function for users to provide feedback.

After reviewing the above existing IF systems, we find that all approaches except InRoute make the assumption that all the features are independent of each other in the process of

learning a profile. Another issue concerned with filtering is uncertainty handling. Sometimes, we cannot be absolutely certain that a document is relevant to a topic since it may be partially relevant to it. An effective text filtering system should be able to take into account the uncertainty. A probabilistic approach can provide a sound theoretic model to solve this problem. It will not simply reject or accept a document but it gives a probability of how likely a document is relevant

To introduce new information system approach based on automatic discretization and Bayesian inference network learning[1]. A Bayesian inference network is based on probability theory and offers a suitable framework for tacklng the uncertainty issue. Features for building profiles are selected automatically to achieve effective filtering. This approach learns profiles expressed as Bayesian inference networks from the training document collection. In contrast, InRoute [3] constructs inference networks from the query or topic. Unlike most existing approaches, our approach does not require the independence assumption of the features. It can adapt to documents of any form and does not need to know in advance the document content as well as the information needs expressed as topics.

## 2.GENERAL ARCHITECTURE [2]

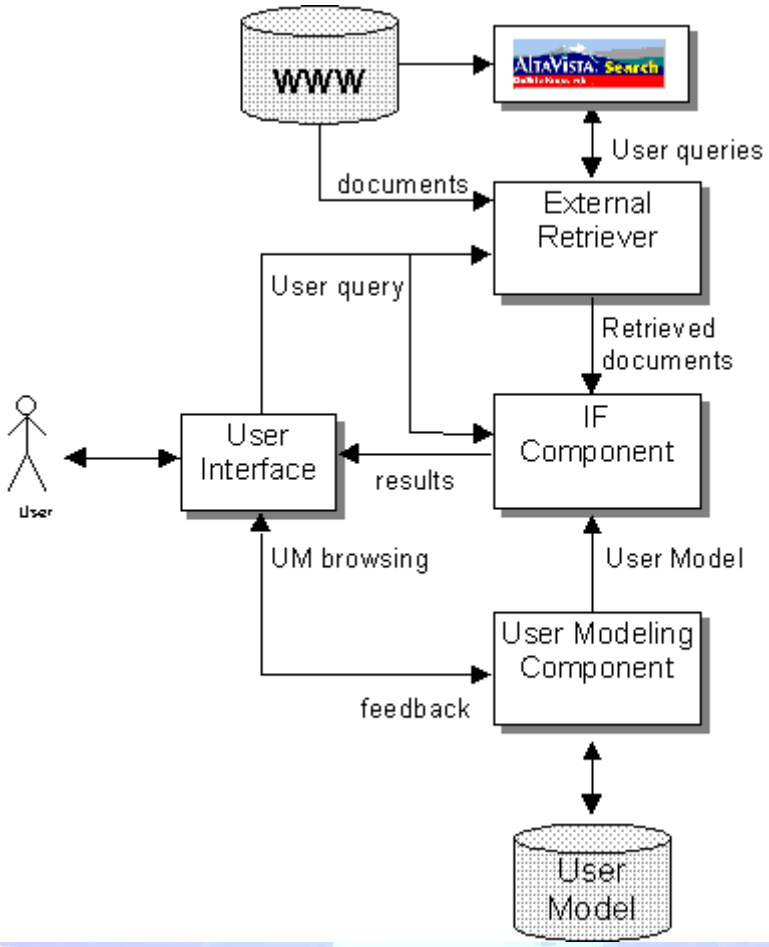


Fig 1.2 Architecture of Intelligent Interface System

- The User Model, representing the characteristics and the information needs of a particular user;
- The User Modeling component, capable of dynamically building the user model, as desumed by the system through the interaction;
- The External Retriever, which interfaces with Anroid;
- The Information Filtering component, which selects the relevant documents for the user, according to the content of the User Model;
- The User Interface, which manages the interaction.

**3.Overview Of Approach**

The main stages of the project are as follows:

1. Data Collection.
2. Data Preprocessing.

3. Profile Generation.
4. Bayesian inference Network.
5. Result of analysis.
6. Ranking

### 3.1 Data Collection:

The most important part while implementing any data related project is collection of proper data for the analysis using any technique (for eg. Data Mining). Thus, in this project collecting some amount of data using search engine As large amount of data is required for implementation of the project collect the requisite amount of data from various sources such as Google search engine or other sources.

### 3.2 Document Preprocessing

#### 3.2.1Document Representation:

First eliminate stop words from the document content. Stop words are words that do not carry useful meaning on its own. Examples of these words are "the," "are," "and," etc. For the remaining words, stemming is applied to them. Stemming is the process of transforming a word into a stem format. For instance, The words "looking" and "looks" are transformed into the same stem "look." Use two kinds of document representation, namely, the word frequency representation (i.e., the number of occurrences) and the word weight representation (inverse document frequency)

The word frequency representation is:

$$Wi = fi \quad \text{----------(1)}$$

The word weight representation is:

$$Wi = fi \ \log(N/ni) \quad \text{----------(2)}$$

#### 3.2.2 Feature Selection:

Typically the total number of terms for a text collection is enormous (e.g., more than 70 000). There is need to use of the whole term set to conduct the task of learning text filters. However, the learning performance will seriously degrade if the feature set is too large. One problem is that the data contains too many irrelevant features which affects the learning process adversely. Another problem is concerned with the computational cost. Usually, the computational resource increases drastically as the number of features increases. One way to alleviate this problem is to

conduct automatic feature selection. Feature selection aims at abstracting the representation of documents to some good features. One of the best techniques for feature selection is expected mutual information measure.

Expected mutual information measure [6] is one of the information- theoretic techniques widely used in pattern recognition and machine learning for feature selection. It measures the degree of association between two elements. In this case, to find features that are strongly associated with the relevance of the topic. Let Cj and C'j denote the fact the document is relevant or irrelevant to the topic, respectively,Wi be a feature and it can take on 0 or 1 representing the absence or presence of the term . The formula for calculating the expected mutual information measure, I(Wi, Cj) , is given as follows:

$$I(Wi,Cj)=\sum_{b=0,1} P(Wi=b,Cj) \log P(Wi=b,Cj)/ P(Wi=b)P(Cj) + P(Wi=b,C'j) \log P(Wi=b,C'j)/ P(Wi=b)P(C'j)$$

where P() denotes a probability. The probabilities can be estimated from the training documents. The higher the expected mutual information measure, the stronger the feature's dependency to the topic is. Select features that have the highest Expected mutual information measure as the predictive features for the topic j. Note that different topics have a different set of feature . Let T'j=(Tj1,….Tjp) denote the p predictive features for the topic j. The value of each Tjk can be a term frequency if use the word frequency representation. This value can be a weight if use the word weight representation.

### 3.3.Feature Discretization

Each document is represented by a feature vector. Both the word weight and the word frequency representation take on a continuous weight value for a feature. So conduct Discretization on each feature. The goal of Discretization is to find a mapping such that the feature value is represented by a discrete value. Suppose we collect all values of the feature f in the training documents and sort the values in ascending order. The mapping is characterized by a series of threshold levels (0, w1,.. .,wk)

Where $0<w1<w2<\ldots<wk$ . Each threshold level is essentially a mid-point of two successive values. Suppose q is a feature value. The mapping has the following property:

$$m(q) = \begin{cases} 0, & \text{if } q=0 \\ i, & \text{if } wi < q < wi+1 \\ k, & \text{if } wk < q \end{cases}$$

Essentially, a pair of consecutive threshold levels define a feature region and all feature values fall into the same region is represented by a unique discrete value. A. Feature value having zero as one region on its own because it represents the case that this feature is absent in the document. The absence of a feature conveys a distinctive meaning which is quite different from other cases. Original feature value is transformed to a different value. Since the objective is to build a classifier and predict the relevance topic by conducting inference, this task only depends on the relationships among the features and the topic. Creating such abstract concepts can, in fact, help revealing the underlying relationships

**3.3.1 Lloyd's Algorithm**

The idea of this algorithm is to minimize the information loss due to discretization. There is a value $\lambda i$ associated with the region I . Each $\lambda i$, which is just the mean of the feature values in the region, serves as a representative value for the region. A distortion metric is defined as taking the square of the difference between the original feature value and the corresponding $\lambda i$. The distortion metric,di , for the region i is given by

$$di = \sum(qi- \lambda i)$$

where qi is a feature value in the region i . To start the Discretization, first select a set of initial threshold levels (y1,…,yk) . These candidate levels are for dividing the whole set of values into regions of values. Given a set of threshold values, the representative value $\lambda i$ of each region is calculated by taking the mean of all the values in a region. Then search for an optimal set of threshold levels based on the distortion metric in each region given in above formula. Clearly, a candidate threshold level should fall between two $\lambda$'s. Test all the threshold levels between the two $\lambda$'s of two regions. In this process, the two $\lambda$'s vary continuously. After testing all the threshold levels within the region, Choose the one that gives the smallest distortion measure for the region that is bounded by a threshold level that is already found and the new threshold level. This process is repeated for other regions.Then, get a new set of threshold levels. check this set of levels with the original set of threshold levels. If the two sets are exactly the same or if the distortion measure for all the regions of the new set is equal or greater than that of the old set,

reject the current set of threshold levels and return the last set found to be the solution. Otherwise, we repeat the above process. The distortion metric for the regions will decrease for every iteration of the above process since the algorithm will converge to stable threshold levels for all the regions.

The following is an example for this Discretization technique. Suppose the feature values are 0.1, 0.1, 0.2, 0.2, 0.3, 0.4, 0.4, 0.5, 0.5, 0.5. divide these values into three regions. First, find two threshold levels to divide the ten values to three regions. choose the level between the third and fourth value as well as between the sixth and seventh value. However, we cannot choose a threshold level that is between two same values, we should either move the position backward or forward until the threshold level is between two different values. Therefore, choose and as the two starting levels. then find the mean of the feature values for each region. For the above example, the mean for the three regions are 0.15, 0.367, and 0.5 respectively. Then,consider the possible threshold levels between the means. For example, if consider the possible threshold levels between 0.15 and 0.367, calculate the mean and the distortion measures of the two affected regions. The level that gives the smallest distortion measure for the first region is chosen as the required threshold. In this case, the level is between 0.2 and 0.3. After finding all the threshold levels, get a new set of threshold levels that has the distortion measures of each region smaller than or equal to that of the old set of levels.

## 4. LEARNING BAYESIAN INFERENCE NETWORKS FOR TEXT FILTERS

objective is to determine whether or not a new document is considered to be relevant to a topic. In essence, to build a classifier for a particular topic. Let Cj denote the fact that the document is relevant to the topic j . Let d be an incoming document to be filtered, technique based on probability theory since it provides a rigorous and formal foundation for handling the uncertainty. The formula for the probability of Cj given according to Bayes' Theorem is as follows:

**P(Cj|d)=P(d|Cj)P(Cj)/P(d)**

To represent the document , use Tj a vector which consists of the predictor features (Tj1,…,Tjp )as discussed in feature selection task above. By substituting d with Tj in above equation, the Bayes rule relates the probability P(Cj|Tj) to P(Tj|Cj) as follows

**P(Cj|Tj)=P(Tj|Cj)P(Cj)/P(Tj)**

If the value of each term in vector Tj is (tj1,…,tjp) , can be rewritten as

**P(Cj|Tj)=P(Tj1=tj1,…,Tjp|Cj)P(Cj)/P(Tj1=tj1,…,Tjp=tjp)**

It is not possible to find enough training documents that have exactly the same feature vector (Tj1=tj1,…Tjp=tjp) to estimate the probabilities P(Tj1=tj1,..,Tjp=tjp|Cj) and P(Tj1=tj1,…Tjp=tjp). However, use the probabilities of each predictor feature to estimate the required probability. A usual approach, such as the naive Bayesian approach is to make the assumption that all the features are independent of each other. And to develop a new approach based on Bayesian inference network induction which relaxes the independence assumptions about the features [1].

### 4.1 Bayesian Networks

A Bayesian inference network is a directed acyclic graph [5],[8] consisting of nodes and arcs. Each node represents a variable which can take on a discrete set of domain specific states. Each arc has a direction. It represents a probabilistic dependency between two nodes. The dependency relationship is represented by the direction of the arc. Specifically the node where the arc arrives depends on the node that the arc comes out.

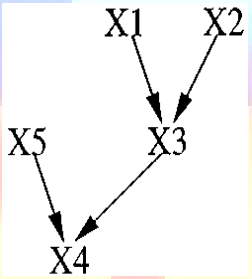


Fig.4.1.1 Example of a Bayesian inference network

The dependency is quantified by a set of conditional probability parameters associated with a node. Let X be a node in the network; $\Upsilon x$ be the set of parents of node X in the network structure.Associated with the node X , there is a conditional probability distribution $P(X|\Upsilon x)$ . If the node X has no parent in the network structure, there is a prior probability distribution P(X) associated with it. An example of a Bayesian inference network is given in Fig. 4.1.1. The above network has five nodes. Suppose each node can take on binary states 0 and 1. Following is an example of the conditional probability distribution of the node X

**P(X3=0|X1=0,X2=0)=0.2**

**P(X3=0|X1=0,X2=1)=0.3**

**P(X3=0|X1=1,X2=0)=0.5**

**P(X3=0|X1=1,X2=1)=0.9**

**P(X3=1|X1=0,X2=0)=0.8**

**P(X3=1|X1=0,X2=1)=0.7**

**P(X3=1|X1=1,X2=0)=0.5**

**P(X3=1|X1=1,X2=1)=0.1**

After the network is constructed, it can be used for conducting reasoning. A common and useful kind of reasoning is to perform probabilistic inferences. The process of inference is to use the evidence of some of the nodes that have observations to find the probability of some of the other nodes in the network.The posterior probability distribution of some other nodes given the observed nodes instantiated with some states

### 4.2 Bayesian Inference Network Learning Approach

To use Bayesian inference networks as classifiers in information filtering problem, There is need to construct a Bayesian inference network to represent the topic profile. This network consists of the set of variables {Cj,Tj1,…Tjp} .Employ a machine learning technique based on our previous work [1] to construct the network automatically from training documents. The network has the predictor features and the relevance as its nodes. The network provides a means to exploit the inherent dependency among these features. After the network is built, we can use it for our filtering task. When a new document arrives, find features that appear in the profile network from the document. Then instantiate appropriately those nodes in the network that represent these features. Probabilistic inferences can then be performed on the network using these instantiations.The posterior probability P(Cj|Tj) is then computed. This probability is compared with a threshold. If the calculated probability exceeds the threshold, the document is considered to be relevant to the topic. By using Bayesian inference networks, relax the independence assumption about features in the profile. The inherent dependency among the features is captured by the network during the learning process.

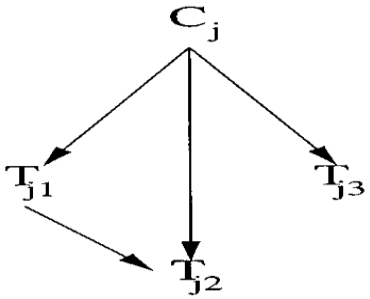


Fig.4 2.1. Example of a classification-based network

**5.Ranking and Result Analysis**

In order to determine whether a document is relevant to a topic, There is need to find the probability of the document being relevant to a topic and set a decision threshold value to determine the relevance of a document. If the document has a probability higher than the decision threshold, the document is assigned to the topic. To design a method called automatic threshold optimization to find an appropriate threshold for a topic. The main procedure for this optimization is as follows. After a model is Learned, use this model to evaluate the training documents. Use the model found to calculate the probability values of the training documents being relevant to a topic. An evaluation measure (to be described below) can then be obtained. Next vary the decision threshold value and repeat the same process. After a number of decision threshold values has been tried, select the one that attains the highest evaluation measure as the decision threshold value for the classifier of that topic. And provide highest ranking to that documents. Rearranging all documents according to new ranking which having highest evaluation measure.

## 6. DATAFLOW DIGRAM

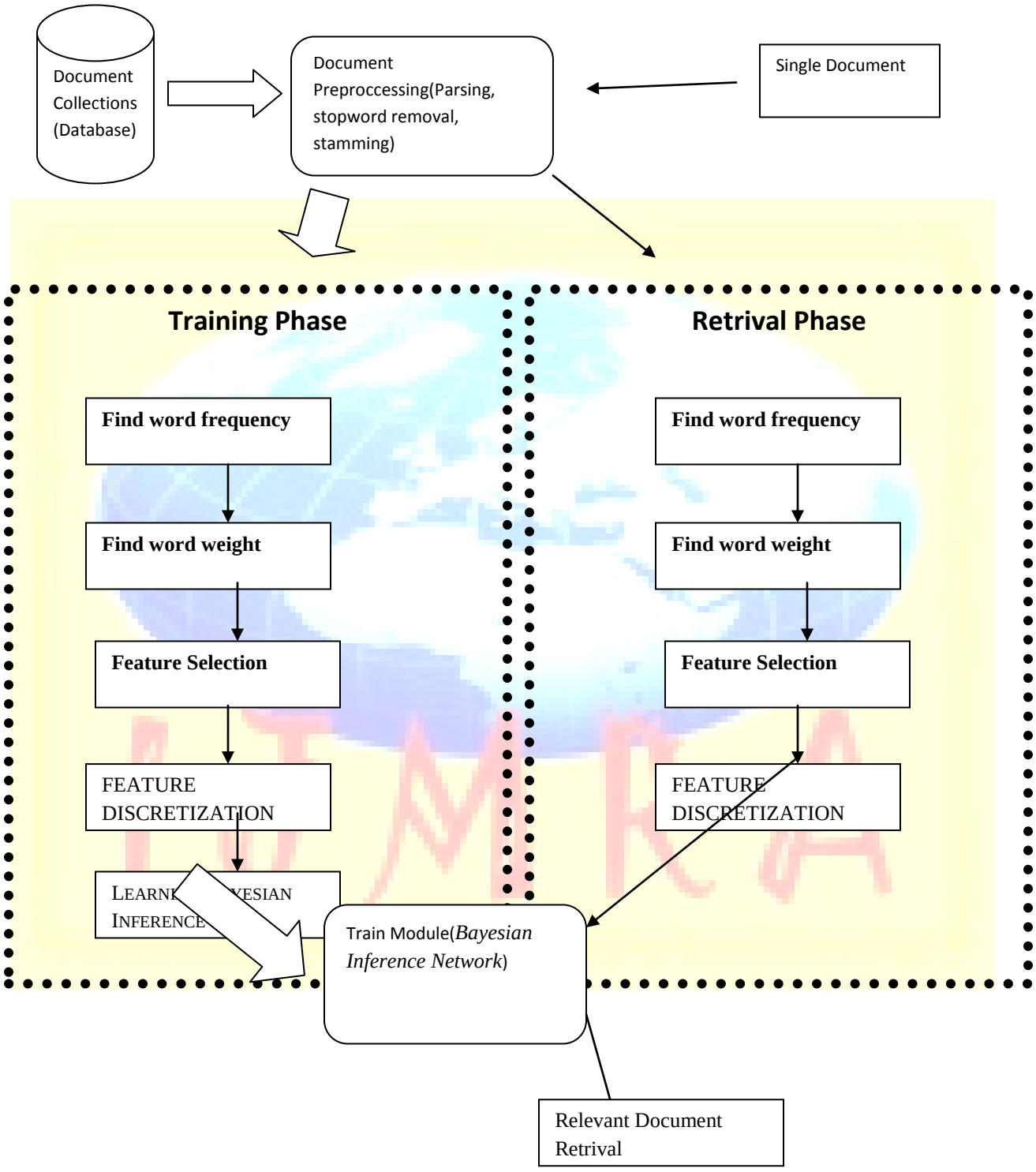


Fig.6.1 Data Flow Of Document Filtering System

REFERENCES


1. Wai Lam and Kon Fan Low Using Discretization and Bayesian Inference Network Learning for Automatic Filtering Profile Generation IEEE transactions on system,man,cybernetics-part C:applications and Reviews,vol 30,No.3 August 2000
2. A Case-Based Approach to Adaptive Information Filtering for the WWW Mauro Marinilli, Alessandro Micarelli and Filippo Sciarrone Dipartimento di Informatica e Automazione Università di Roma Tre Via della Vasca Navale, 79 I00146 Roma, Italia
3. J. Callan, "Document filtering with inference networks," in Proc. 19th Int. ACM SIGIR Conf. Research Development Information Retrieval,1996, pp. 262–269.
4. N. J. Belkin and B. W. Croft, "Information filtering and information retrieval: Two sides of the same coin?," Commun. ACM, vol. 35, no. 12, pp. 29–38, 1992.
5. E. Charniak, "Bayesian networks without tears," AI Mag., pp. 50–63,Winter 1991.
6. R. Fano, Transmission of Information. Cambridge, MA: MIT Press,1961.
7. N. Fuhr and U. Pfeifer, "Probabilistic information retrieval as a combination of abstraction, inductive learning and probabilistic assumptions,"ACM Trans. Inf. Syst., vol. 12, no. 1, pp. 92–115, 1994.retrieval," Commun. ACM, vol. 38, no. 3, pp. 42–48, 57, 1995.
8. D. Heckerman and M. P. Wellman, "Bayesian networks," Commun.ACM, vol. 38, no. 3, pp. 27–41, 1995.
9. W. Lam and F. Bacchus, "Learning Bayesian belief networks. An approach based on the MDL principal," Comput. Intell., vol. 10, no. 3, pp.269–293, 1994.
10. W. Lam and K. F. Low, "Constructing text filters based on Bayesian network learning," in Proc. 13th Eur. Conf. Artificial Intelligence, 1998,pp. 585–589.
11. W. Lam, K. F. Low, and C. Y. Ho, "Using a Bayesian network induction approach for text categorization," in Proc. 15th Int. Joint Conf. Artificial Intelligence, 1997, pp. 23–29.
12. W. Lam, S. Mukhopadhyay, J. Mostafa, and M. Palakal, "Detection of shifts in user interests for personalized information filtering," in Proc. 19th Int. ACM SIGIR Conf. Research Development Information Retrieval, 1996, pp. 317–325.
13. W. Lam, M. Ruiz, and P. Srinivasan, "Automatic text categorization andits application to text retrieval," IEEE Trans. Knowl. Data Eng., vol. 11,no. 6, pp. 865–879, 1999.

14. K. Lang, "Newsweeder: Learning to filter netnews," in Proc. Int. Conf. Machine Learning, 1995, pp. 331–339.
15. M. Pazzani and D. Billsus, "Learning and revising user profiles: The identification of interesting web sites," Mach. Learn., vol. 27, no. 3, pp. 313–331, 1997.
16. C. J. Van Rijsbergen, Information Retrieval. London, U.K.: Butterworth, 1979.
17. G. Salton and M. J. McGill, Introduction to Modern Information Rerieval. New York: McGraw-Hill, 1983.
18. B. Sheth, "A learning approach to personalized information filtering,"M.S. thesis, Dept. Elect. Eng. Comput. Sci., Mass. Inst. Technol., Cambridge, 1994.
19. B. Sheth and P. Maes, "Evolving agents for personalized information filtering," in Proc. 9th Conf. Artificial Intelligence Applications, 1993.
20. K. Tzeras and S. Hartmann, "Automatic indexing based on Bayesian inference networks," in Proc. Int. ACMSIGIR Conf. Res. Develop. Inform. Retrieval, 1993.
21. S. K. M. Wong and Y. Y. Yao, "On modeling information retrieval with probabilistic inference," ACMTrans. Inf. Syst., vol. 13, no. 1, pp. 38–68, 1995.
22. S. K. M. Wong and W. Ziarko, "A machine learning approach in information retrieval," in Proc. Int. ACM SIGIR Conf. Res. Develop. Inform. Retrieval, 1986, pp. 228–233.
23. T. W. Yan and H. Garcia-Molina, "Sift—A tool for wide-area information dissemination," in Proc. 1995 USENIX Tech. Conf., 1995, pp. 177–186
24. R. Fung and B. Del Favero, "Applying Bayesian networks to information retrieval," Commun. ACM, vol. 38, no. 3, pp. 42–48, 57, 1995.