Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval

David D. Lewis

AT&T Labs — Research
180 Park Avenue
Florham Park, NJ 07932-0971 USA
lewis@research.att.com
http://www.research.att.com/~lewis

Abstract. The naive Bayes classifier, currently experiencing a renaissance in machine learning, has long been a core technique in information retrieval. We review some of the variations of naive Bayes models used for text retrieval and classification, focusing on the distributional assumptions made about word occurrences in documents.

1 Introduction

The naive Bayes classifier, long a favorite punching bag of new classification techniques, has recently emerged as a focus of research itself in machine learning. Machine learning researchers tend to be aware of the large pattern recognition literature on naive Bayes, but may be less aware of an equally large information retrieval (IR) literature dating back almost forty years [37, 38]. In fact, naive Bayes methods, along with prototype formation methods [44, 45, 24], accounted for most applications of supervised learning to information retrieval until quite recently.

In this paper we briefly review the naive Bayes classifier and its use in information retrieval. We concentrate on the particular issues that arise in applying the model to textual data, and provide pointers to the relevant IR and computational linguistics literature. We end with a few thoughts on interesting research directions for naive Bayes at the intersection of machine learning and information retrieval.

2 The Naive Bayes Classifier

A widely used framework for classification is provided by a simple theorem of probability [10, Sec 2.1] known as *Bayes' rule*, *Bayes' theorem*, or *Bayes' formula*:

$$P(C = c_k|\mathbf{X} = \mathbf{x}) = P(C = c_k) \times \frac{P(\mathbf{X} = \mathbf{x}|C = c_k)}{P(\mathbf{x})} \qquad (1)$$

where

$$P(\mathbf{X}=\mathbf{x}) = \sum_{k'=1}^{e_C} P(\mathbf{X}=\mathbf{x}|C=c_{k'}) \times P(C=c_{k'}) \tag{2}$$

We assume here that all possible events (in our case, documents) fall into exactly one of e_C classes, $(c_1, \ldots, c_k, \ldots, c_{e_C})$. C is a random variable whose values are those classes, while $\mathbf{X}$ is a vector random variable whose values are vectors of feature values $\mathbf{x} = (x_1, \ldots, x_j, \ldots, x_d)$, one vector for each document. (Except where stated, we will assume that $\mathbf{x}$ has the same length, d, for each document.) $P(C = c_k|\mathbf{X} = \mathbf{x})$ is the conditional probability that a document belongs to class c_k, given that we know it has feature vector $\mathbf{x}$. Bayes' rule specifies how this conditional probability can be computed from the conditional probabilities of seeing particular vectors of feature values for documents of each class, and the unconditional probability of seeing a document of each class.

Having made clear that c_k and $\mathbf{x}$ are values taken on by random variables C and $\mathbf{X}$ we simplify notation by omitting those random variables and instead writing Bayes' rule as:

$$P(c_k|\mathbf{x}) = P(c_k) \times \frac{P(\mathbf{x}|c_k)}{P(\mathbf{x})} \tag{3}$$

When we know the $P(c_k|\mathbf{x})$ exactly for a classification problem, classification can be done in an optimal way for a wide variety of effectiveness measures [10, 31]. For instance, the expected number of classification errors can be minimized by assigning a document with feature vector $\mathbf{x}$ to the class c_k for which $P(c_k|\mathbf{x})$ is highest.

We of course do not know the $P(c_k|\mathbf{x})$ and must estimate them from data, which is difficult to do directly. Bayes' rule suggests instead estimating $P(\mathbf{x}|c_k)$, $P(c_k)$, and $P(\mathbf{x})$, and then combining those estimates to get an estimate of $P(c_k|\mathbf{x})$. However, even estimating the $P(\mathbf{x}|c_k)$ poses problems, since there are usually an astronomical number of possible values for $\mathbf{x} = (x_1, \ldots, x_j, \ldots, x_d)$. A common strategy is to assume that the distribution of $\mathbf{x}$ conditional on c_k can be decomposed in this fashion for all c_k:

$$P(\mathbf{x}|c_k) = \prod_{j=1}^{d} P(x_j|c_k) \tag{4}$$

The assumption here is that the occurrence of a particular value of x_j is statistically independent of the occurrence of any other $x_{j'}$, given that we have a document of type c_k. The advantage of making this assumption is that we typically can model the $P(x_j|c_k)$ with relatively few parameters.

If we assume Equation 4, then Equation 3 becomes:

$$P(c_k|\mathbf{x}) = P(c_k) \times \frac{\prod_{j=1}^{d} P(x_j|c_k)}{P(\mathbf{x})} \tag{5}$$

where we now have

$$P(\mathbf{x}) = \sum_{k'=1}^{e_C} P(c_{k'}) \times \prod_{j'=1}^{d} P(x_{j'}|c_{k'}). \tag{6}$$

If we plug in estimates (indicated by carats) for the values on the right hand side, we get an estimate for $P(c_k|\mathbf{x})$:

$$\widehat{P(c_k|\mathbf{x})} = \frac{\widehat{P(c_k)} \times \prod_{j=1}^{d} \widehat{P(x_j|c_k)}}{\widehat{P(\mathbf{x})}} \tag{7}$$

This estimate can then be used for classification. If the goal of classification is to minimize number of errors, then we can assign a document with feature vector $\mathbf{x}$ to the c_k such that $\widehat{P(c_k|\mathbf{x})}$ is highest. A classifier which operates in this fashion is sometimes known as a *naive Bayes classifier*. Typically the denominator in Equation 7 is not explicitly computed for minimum-error classification, since it is the same for all c_k. Instead the maximum value of the numerator is found as used to make a classification decision.

Indeed, classification will be accurate as long as the correct class has the highest value of $\widehat{P(c_k)} \times \prod_{j=1}^{d} \widehat{P(x_j|c_k)}$, regardless of whether that is a good estimate of $P(c_k|\mathbf{x})$ (Section 6).

3 Text Representation

Before discussing the classification of documents using naive Bayes, we must say a bit about what a document is, and how it is represented. A document is typically stored as a sequence of characters, with characters representing the text of a written natural language expression.[1] Information retrieval has developed a variety of methods for transforming the character string representing a document into a form more amenable to statistical classification. These methods are analogous to, if less complex than, the feature extraction methods used in speech recognition, image processing, and related disciplines.

A wide variety of statistical, linguistic, and knowledge-based techniques, involving various amounts of machine and/or manual processing, have been used to produce representations of text for information retrieval systems ([12, Chs. 7-9], [28, Ch. 5], [29, Chs. 3-6] [46, Ch. 3], [51, Chs. 2-3]). An ongoing surprise and disappointment is that structurally simple representations produced without linguistic or domain knowledge have been as effective as any others [30, 33]. We therefore make the common assumption that the preprocessing of the document produces a bag (multiset) of *index terms* which do not themselves have internal structure. This representation is sometimes called the *bag of words* model. For the purposes of our discussion, it does not matter whether the index terms

[1] More generally a document may have various components (title, body, sections, etc.) which are, for the most part, pieces of text. We will concentrate here on the simplest case, where the document is a single piece of text.

are actually words, or instead character n-grams, morphemes, word stems, word n-grams, or any of a number of similar text representations.

4 The Binary Independence Model

Having reduced the richness of language to a bag of symbols, information retrieval commonly goes even farther. Suppose we have a collection of documents and associate a binary feature x_j with each of the d unique words we observe in the collection. The feature will equal 1 if the corresponding word occurs in the document, and 0 otherwise. The full document representation then is $\mathbf{x} = (x_1, \ldots, x_j, \ldots, x_d)$, where all x_j are 0 or 1.

If we make the naive Bayes assumption of conditional independence of feature values given class membership, then the conditional probability of observing feature vector $\mathbf{x}$ for a document of class c_k is given by Equation 5. However, the combination of binary features with the conditional independence assumption allows an even simpler expression for the posterior probability. Note that:

$$P(x_j|c_k) = p_{jk}^{x_j}(1-p_{jk})^{1-x_j} \tag{8}$$
$$= \left(\frac{p_{jk}}{1-p_{jk}}\right)^{x_j}(1-p_{jk}) \tag{9}$$

where $p_{jk} = P(x_j = 1|c_k)$. Using this fact in combination with Equation 5 and some rearranging of terms[2] we get:

$$\log P(c_k|\mathbf{x}) = \log P(c_k) + \sum_{j=1}^{d} x_j \log \frac{p_{jk}}{1-p_{jk}} + \sum_{j=1}^{d} \log(1-p_{jk}) - \log P(\mathbf{x}) \tag{10}$$

for each c_k. Except for $\log P(\mathbf{x})$, this has the convenient property of being a linear function of the feature values. In many uses of naive Bayes, however, we care only which $P(c_k|\mathbf{x})$ (or $\log P(c_k|\mathbf{x})$) is largest, not their exact values. In that case, we can drop the $\log P(\mathbf{x})$, since it is the same for all c_k.

It is common in information retrieval that we have only two classes between which we wish to discriminate. In text retrieval, we want to separate those documents relevant to a user of a search engine from those not relevant. In text categorization we often need to decide only whether a document should be assigned to a particular subject category or not. In the two class case, we have $P(c_2|\mathbf{x}) = 1 - P(c_1|\mathbf{x})$, so that with some arithmetic manipulations we can replace the two functions that Equation 10 would give for the two-class case with a single function [10, Sec. 2.10]:

[2] See [39, Sec. 12.4.3] or [10, Sec. 2.10], though in their derivations $P(\mathbf{x})$ has already been dropped.

$$\log \frac{P(c_1|\mathbf{x})}{1 - P(c_1|\mathbf{x})} = \sum_{j=1}^{d} x_j \log \frac{p_{j1}(1 - p_{j2})}{(1 - p_{j1})p_{j2}} + \sum_{j=1}^{d} \log \frac{1 - p_{j1}}{1 - p_{j2}} + \log \frac{P(c_1)}{1 - P(c_1)}. \quad (11)$$

Equation 11 has several properties that make it particularly convenient. First, we observe that $\log \frac{P(c_1|\mathbf{x})}{1-P(c_1|\mathbf{x})}$ is monotonic with (and if necessary can be used to compute) $P(c_1|\mathbf{x})$. It therefore suffices for any purpose which we might use $P(c_1|\mathbf{x})$ for. Second, the equation is truly linear in the x_j (since $P(\mathbf{x})$ disappears completely in the two-class case), and has only $d+1$ parameters to estimate and store.

A further advantage in the context of information retrieval is that Equation 11 requires *presence weights* only. That is, if one sets the initial score of a document to be the constant term in Equation 11, the full score can be computed by adding up values involving only those words present in a document, not those absent from the document [41,48]. Since most words do not occur in most documents, this is desirable from the standpoint of computational efficiency.

The two-class, binary feature naive Bayes model has come to be known in information retrieval as the *binary independence model*. Its use in the form of Equation 11 was promoted by Robertson and Sparck Jones in a paper [41] that did much to clarify and unify a number of related and partially ad hoc applications of naive Bayes dating back to Maron [37].

Robertson and Sparck Jones' particular interest in the binary independence model was its use in *relevance feedback* [20,45]. In relevance feedback, a user query is given to a search engine, which produces an initial ranking of its document collection by some means. The user examines the initial top-ranked documents and gives feedback to the system on which are relevant to their interest and which are not. The search engine then applies supervised learning to these judgments to produce a formula that can be used to rerank the documents.

Robertson and Sparck Jones noted that if a system does not need to choose between c_1 and c_2, but only to rank documents in order of $P(c_1|\mathbf{x})$, then the only quantity needed from Equation 11 is:

$$\sum_{j=1}^{d} x_j \log \frac{p_{j1}(1 - p_{j2})}{(1 - p_{j1})p_{j2}}. \quad (12)$$

All other values in Equation 11 are constant across $\mathbf{x}$'s, and so can dropped. the result is still monotonic with $P(c_1|\mathbf{x})$, but does not require an estimate of the prior $P(c_1)$. Such an estimate is difficult to obtain either from users or from the small, nonrandom samples available for training in a relevance feedback context.

4.1 Weaknesses of the BIM

While the BIM has been very influential in information retrieval, it has shortcomings that mean it is now rarely used in the pure form given above. One weakness is that by considering only the presence or absence of terms, the BIM ignores information inherent in the frequencies of terms. For instance, all things being equal, we would expect that if 1 occurrence of a word is a good clue that a document belongs to a class, then 5 occurrences should be even more predictive.

A related problem concerns document length. As a document gets longer, the number of distinct words used, and thus the number of values of x_j that equal 1 in the BIM, will in general increase. Many of these word usages in very long documents will be unrelated to the core content of the document, but are treated as being of the same significance as similar occurrences in short documents. Again, all things being equal, we would expect that 1 occurrence of a good predictor in a short document is a better clue than 1 occurrence of that predictor in a long document.

Ignoring document length can have a particularly bad interaction with feature selection. It is common in IR that one class (let's say c_1) is of much lower frequency than its contrasting class (c_2). The class $c_1$might be those documents relevant to a user (vs. the much larger class of nonrelevant documents), or the class of documents on a particular subject (vs. all those not on the subject). Further, it is common that most words have skewed frequencies as well, resulting in binary features that much more often take on a value of 0 than 1. In this situation, typical feature selection measures strongly prefer features correlated with c_1, so that we often have:

$$\log \frac{p_{j1}(1 - p_{j2})}{(1 - p_{j1})p_{j2}} > 0 \tag{13}$$

for all selected features. Some feature selection measures used in IR in fact explicitly require features to have this property. When all features have this property, increasing document length can only increase the estimate $\widehat{P(c_k|\mathbf{x})}$, regardless of the actual content of the document. While a case can be made that longer documents are somewhat more likely to be of interest to any given user [43,47], the above effect is likely to be far stronger than appropriate.

5 Other Distributional Models

In this section we look at a number of variations on the naive Bayes model that attempt to address the weaknesses of the BIM.

5.1 Distributions for Integer-Valued Features

The most straightforward generalization of the BIM is to let the X_j be integer-valued random variables corresponding to *term frequencies*, that is counts of the number of occurrences of words in a document. The naive Bayes model will still assume the X_j are independently distributed, but now each is modeled by an integer-valued distribution rather than a Bernoulli one.

A variety of statistical distributions for term frequencies have been investigated, some in the context of naive Bayes classifiers and some for other purposes. The distributions investigated have mostly been Poisson mixtures [4, 26]: the Poisson itself [40], mixtures of 2, 3, or more Poissons [1, 2, 22, 23, 36], and the negative binomial (an infinite mixture of Poissons) [40]. The details of the particular models can be complex, sometimes involving latent variables that intervene between the class label and the term frequencies. Rather than attempt to survey the variations here, we refer the reader to the above references, with the suggestion that the book by Mosteller and Wallace [40] is the most clear treatment from a classification standpoint.

Despite considerable study, explicit use of Poisson mixtures for text retrieval have not proven more effective than using the BIM [35, 42]. This failure has been variously blamed on the larger number of parameters these models require estimating, the choice of estimation methods, the difficulty of accounting for document length in these models, and the poor fit of the models to actual term frequencies. In contrast, a recently proposed term weighting formula which rescales the BIM weight to in some ways approximate the behavior of a two-Poisson model has proven quite successful [43]. It should be noted, however, that most studies of Poisson mixtures (Mosteller and Wallace being an exception) have been applications to text retrieval rather than routing or categorization (where more training data is available), and/or have focused on unsupervised fitting of Poisson mixtures rather than supervised learning with a naive Bayes model.

5.2 Multinomial Models

An alternative approach to modeling term frequencies is to treat the bag of words for a length f document as resulting from f draws on a d-valued multinomial variable $\mathbf{X}$, rather than as a single draw on a vector-valued variable of length d [15]. The naive Bayes assumption then is that the draws on $\mathbf{X}$ are independent—each word of the document is generated independently from every other.

A multinomial model has the advantage that document length is accounted for very naturally in the model. The corresponding disadvantage is that it assumes independence not just between different words, but between multiple occurrences of the same word, an assumption which is strikingly violated for real data [4]. A multinomial therefore assigns extreme posterior log odds to long documents, and would presumedly be very poor for the purpose of ranking documents in a search engine. The problem is somewhat less extreme for classification tasks, where we can in some cases arrange to compare posterior log odds of classes for each document individually, without comparisons across documents. Indeed, we know of many applications of multinomial models to text categorization [3, 14, 15, 25, 32, 34] but none to text retrieval.

5.3 Non-Distributional Approaches

A variety of ad hoc approaches have been developed that more or less gracefully integrate term frequency and document length information into the BIM itself. The widely used *probabilistic indexing* approach assumes there is an ideal binary indexing of the document, for which the observed index term occurrences provide evidence [7, 13]. Retrieval or classification is based on computing (or approximating) the expected value of the posterior log odds. The expectation is taken with respect to the probabilities of various ideal indexings. While this is a plausible approach, in practice the probabilities of the ideal indexings are computed by ad hoc functions of term frequency, document length, and other quantities, making these models not truly distributional.

Another approach is to fit a distributional but nonparametric model (for instance a linear regression) to predict the probability that a given term frequency will be observed in a document of a particular length [53]. Such nonparametric approaches have been relatively rare in IR, and it appears that the sophisticated discretization and kernel based approaches investigated in machine learning have not been tried.

6 Violated Assumptions and the Success of Naive Bayes

As has often been observed, the independence assumptions on which naive Bayes classifiers are based almost never hold for natural data sets, and certainly not for textual data. This contradiction has motivated three kinds of research in both information retrieval and machine learning: 1) attempts to produce better classifiers by relaxing the independence assumption, 2) modifications of feature sets to make the independence assumption more true, and 3) attempts to explain why the independence assumption isn't really needed anyway.

Whatever its successes in machine learning, the first strategy has not met with great success in IR. While interesting research on dependence models has been done [8, 11, 21, 49, 50], these models are rarely used in practice. Even most work in the "inference net" approach to information retrieval has mostly used independence (or ad hoc) models.

Results from the second strategy are hard to judge. A variety of text representation strategies which tend to reduce independence violations have been pursued in information retrieval, including stemming, unsupervised term clustering, downcasing of text, phrase formation, and feature selection. However, these strategies have usually been pursued for reasons besides reducing feature dependence, and so there has been little attempt to correlate their actual impact on dependence with any effectiveness changes they yield. Further, the nature of this impact is more complex than might be guessed, even for very simple techniques [4]. In any case, the effectiveness improvements yielded by these strategies have been small (with the possible selection of feature selection).

IR's representative of the third strategy is Cooper [6], who points out that in the case of a two-class naive Bayes model, the usual independence assumptions (Equation 4) can be replaced by a weaker "linked dependence" assumption:

$$\frac{P(\mathbf{x}|c_1)}{P(\mathbf{x}|c_2)} = \prod_{j=1}^{d} \frac{P(x_j|c_1)}{P(x_j|c_2)} \tag{14}$$

In machine learning, considerable theoretical and experimental evidence has been developed that a training procedure based on the naive Bayes assumptions can yield an optimal classifier in a variety of situations where the assumptions are wildly violated [9].

7 Conclusion

Naive Bayes models have been remarkably successful in information retrieval. In the yearly TREC evaluations [16–19, 52], numerous variations of naive Bayes models have been used, producing some of the best results. Recent comparisons of learning methods for text categorization have been somewhat less favorable to naive Bayes models [5, 25] while still showing them to achieve respectable effectiveness. This may be because the larger amount of training data available in text categorization data sets favors algorithms which produce more complex classifiers [27], or may because the more elaborate representation and estimation tricks developed for retrieval and routing with naive Bayes have not been applied to categorization.

There are many open research questions on the application of naive Bayes in information retrieval. What is a reasonable distributional model taking into account term frequency and document length? Can we state necessary or sufficient conditions for when a naive Bayes model will produce an optimal ranking of documents? What is the optimal strategy for selecting training data for naive Bayes? And, of course, can dependence information actually be used to improve the effectiveness of naive Bayes classifiers? These and other questions will provide great interest for both machine learning and information retrieval in the years to come.

References

1. Abraham Bookstein and Don Kraft. Operations research applied to document indexing and retrieval decisions. *Journal of the Association for Computing Machinery*, 24(3):418–427, 1977.
2. Abraham Bookstein and Don R. Swanson. A decision theoretic foundation for indexing. *Journal of the American Society for Information Science*, pages 45–50, January-February 1975.
3. Soumen Chakrabarti, Byron Dom, Rakesh Agrawal, and Prabhakar Raghavan. Using taxonomy, discriminants, and signatures for navigating in text databases. In Matthias Jarke, Michael Carey, Klaus R. Dittrich, Fred Lochovsky, Pericles Loucopoulos, and Manfred A. Jeusfeld, editors, *Proceedings of the 23rd VLDB Conference*, pages 446–455, 1997.
4. Kenneth Ward Church. One term or two? In Edward A. Fox, Peter Ingwersen, and Raya Fidel, editors, *SIGIR '95: Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 310–318, New York, 1995. Association for Computing Machinery.
5. William W. Cohen and Yoram Singer. Context-sensitive learning methods for text categorization. In *SIGIR '96: Proceedings of the 19th Annual International ACM*

SIGIR Conference on Research and Development in Information Retrieval, pages 307–315, 1996.
6. W. S. Cooper. Some inconsistencies and misidentified modeling assumptions in probabilistic information retrieval. *ACM Transactions on Information Systems*, 13(1):100–111, January 1995.
7. W. B. Croft. Experiments with representation in a document retrieval system. *Information Technology: Research and Development*, 2:1–21, 1983.
8. W. Bruce Croft. Boolean queries and term dependencies in probabilistic retrieval models. *Journal of the American Society for Information Science*, 37(2):71–77, 1986.
9. Pedro Domingos and Michael Pazzani. On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning*, 29(2/3):103–130, November 1997.
10. Richard O. Duda and Peter E. Hart. *Pattern Classification and Scene Analysis.* Wiley-Interscience, New York, 1973.
11. B. Del Favero and R. Fung. Bayesian inference with node aggregation for information retrieval. In D. K. Harman, editor, *The Second Text Retrieval Conference (TREC-2)*, pages 151–162, Gaithersburg, MD, March 1994. U. S. Dept. of Commerce, National Institute of Standards and Technology. NIST Special Publication 500-215.
12. William B. Frakes and Ricardo Baeza-Yates, editors. *Information Retrieval: Data Structures and Algorithms.* Prentice Hall, Englewood Cliffs, NJ, 1992.
13. Norbert Fuhr. Models for retrieval with probabilistic indexing. *Information Processing and Management*, 25(1):55–72, 1989.
14. William A. Gale, Kenneth W. Church, and David Yarowsky. A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, 26:415–439, 1993.
15. Louise Guthrie, Elbert Walker, and Joe Guthrie. Document classification by machine: Theory and practice. In *COLING 94: The 15th International Conference on Computational Linguistics. Proceedings, Vol. II.*, pages 1059–1063, 1994.
16. D. K. Harman, editor. *The First Text REtrieval Conference (TREC-1)*, Gaithersburg, MD 20899, 1993. National Institute of Standards and Technology. Special Publication 500-207.
17. D. K. Harman, editor. *The Second Text REtrieval Conference (TREC-2)*, Gaithersburg, MD 20899, 1994. National Institute of Standards and Technology. Special Publication 500-215.
18. D. K. Harman, editor. *Overview of the Third Text REtrieval Conference (TREC-3)*, Gaithersburg, MD 20899-0001, 1995. National Institute of Standards and Technology. Special Publication 500-225.
19. D. K. Harman, editor. *The Fourth Text REtrieval Conference (TREC-3)*, Gaithersburg, MD 20899-0001, 1996. National Institute of Standards and Technology. Special Publication 500-236.
20. Donna Harman. Relevance feedback and other query modification techniques. In William B. Frakes and Ricardo Baeza-Yates, editors, *Information Retrieval: Data Structures and Algorithms*, pages 241–263. Prentice Hall, Englewood Cliffs, NJ, 1992.
21. D. J. Harper and C. J. van Rijsbergen. An evaluation of feedback in document retrieval using co-occurrence data. *Journal of Documentation*, 34:189–216, 1978.
22. Stephen P. Harter. A probabilistic approach to automatic keyword indexing. Part I. On the distribution of specialty words in a technical literature. *Journal of the American Society for Information Science*, pages 197–206, July-August 1975.

23. Stephen P. Harter. A probabilistic approach to automatic keyword indexing. Part II. An algorithm for probabilistic indexing. *Journal of the American Society for Information Science*, pages 280–289, September-October 1975.
24. David J. Ittner, David D. Lewis, and David D. Ahn. Text categorization of low quality images. In *Symposium on Document Analysis and Information Retrieval*, pages 301–315, Las Vegas, NV, 1995. ISRI; Univ. of Nevada, Las Vegas.
25. Thorsten Joachims. Text categorization with support vector machines: Learning with many relevant features. LS-8 Report 23, University of Dortmund, Computer Science Dept., Dortmund, Germany, 27 November 1997.
26. S. Katz. Distribution of content words and phrases in text and language modelling. *Natural Language Engineering*, 2(1):15–59, March 1996.
27. Ron Kohavi. Scaling up the accuracy of Naive-Bayes classifiers: a decision-tree hybrid. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pages 202–207, 1996.
28. Robert R. Korfhage. *Information Storage and Retrieval.* John Wiley, New York, 1997.
29. Gerald Kowalski. *Information Retrieval Systems: Theory and Implementation.* Kluwer, Boston, 1997.
30. David D. Lewis. Text representation for intelligent text retrieval: A classification-oriented view. In Paul S. Jacobs, editor, *Text-Based Intelligent Systems*, pages 179–197. Lawrence Erlbaum, Hillsdale, NJ, 1992.
31. David D. Lewis. Evaluating and optimizing autonomous text classification systems. In Edward A. Fox, Peter Ingwersen, and Raya Fidel, editors, *SIGIR '95: Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 246–254, New York, 1995. Association for Computing Machinery.
32. David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers. In W. Bruce Croft and C. J. van Rijsbergen, editors, *SIGIR 94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 3–12, London, 1994. Springer-Verlag.
33. David D. Lewis and Karen Sparck Jones. Natural language processing for information retrieval. *Communications of the ACM*, 39(1):92–101, January 1996.
34. Hang Li and Kenji Yamanishi. Document classification using a finite mixture model, 1997.
35. Robert M. Losee. Parameter estimation for probabilistic document-retrieval models. *Journal of the American Society for Information Science*, 39(1):8–16, 1988.
36. E. L. Margulis. Modelling documents with multiple Poisson distributions. *Information Processing and Management*, 29:215–227, 1993.
37. M. E. Maron. Automatic indexing: An experimental inquiry. *Journal of the Association for Computing Machinery*, 8:404–417, 1961.
38. M. E. Maron and J. L. Kuhns. On relevance, probabilistic indexing, and information retrieval. *Journal of the Association for Computing Machinery*, 7(3):216–244, July 1960.
39. Marvin Minsky and Seymour Papert. *Perceptrons: An Introduction to Computational Geometry (Expanded Edition).* The MIT Press, Cambridge, MA, 1988.
40. Frederick Mosteller and David L. Wallace. *Applied Bayesian and Classical Inference.* Springer-Verlag, New York, 2nd edition, 1984.
41. S. E. Robertson and K. Sparck Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, pages 129–146, May-June 1976.

42. S. E. Robertson, C. J. van Rijsbergen, and M. F. Porter. Probabilistic models of indexing and searching. In R. N. Oddy, S. E. Robertson, C. J. van Rijsbergen, and P. W. Williams, editors, *Information Research and Retrieval*, chapter 4, pages 35–56. Butterworths, 1981.
43. S. E. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In W. Bruce Croft and C. J. van Rijsbergen, editors, *SIGIR 94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pages 232–241, London, 1994. Springer-Verlag.
44. J. J. Rocchio, Jr. Relevance feedback in information retrieval. In Gerard Salton, editor, *The SMART Retrieval System: Experiments in Automatic Document Processing*, pages 313–323. Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1971.
45. Gerard Salton and Chris Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4):288–297, 1990.
46. Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill Book Company, New York, 1983.
47. Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted document length normalization. In *SIGIR '96: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 21–29, 1996.
48. Karen Sparck Jones. Search term relevance weighting given little relevance information. *Journal of Documentation*, 35(1):30–48, March 1979.
49. Howard R. Turtle and W. Bruce Croft. Evaluation of an inference network-based retrieval model. *ACM Transactions on Information Systems*, 9(3):187–222, July 1991.
50. C. J. van Rijsbergen. A theoretical basis for the use of co-occurrence data in information retrieval. *Journal of Documentation*, 33(2):106–119, June 1977.
51. C. J. van Rijsbergen. *Information Retrieval*. Butterworths, London, second edition, 1979.
52. E. M. Voorhees and D. K. Harman, editors. *Information Technology: The Fifth Text REtrieval Conference (TREC-5)*, Gaithersburg, MD 20899-0001, 1997. National Institute of Standards and Technology. Special Publication 500-238.
53. Clement T. Yu and Hirotaka Mizuno. Two learning schemes in information retrieval. In *Eleventh International Conference on Research & Development in Information Retrieval*, pages 201–215, 1998.