Predicting the Effectiveness of Naïve Data Fusion on the Basis of System Characteristics

Kwong Bor Ng
Graduate School of Library and Information Studies, Queens College, City University of New York, Rosenthal Library Room 254, Flushing, NY 11367. E-mail: kbng@qc.edu

Paul B Kantor
APLab, SCILS, Rutgers University, New Brunswick, NJ 08903. E-mail: kantor@scils.rutger.sdu

Effective automation of the information retrieval task has long been an active area of research, leading to sophisticated retrieval models. With many IR schemes available, researchers have begun to investigate the benefits of combining the results of different IR schemes to improve performance, in the process called "data fusion." There are many successful data fusion experiments reported in IR literature, but there are also cases in which it did not work well. Thus, if would be quite valuable to have a theory that can predict, in advance, whether fusion of two or more retrieval schemes will be worth doing. In previous study (Ng & Kantor, 1998), we identified two predictive variables for the effectiveness of fusion: (a) a list-based measure of output dissimilarity, and (b) a pair-wise measure of the similarity of performance of the two schemes. In this article we investigate the predictive power of these two variables in simple symmetrical data fusion. We use the IR systems participating in the TREC 4 routing task to train a model that predicts the effectiveness of data fusion, and use the IR systems participating in the TREC 5 routing task to test that model. The model asks, "when will fusion perform better than an oracle who uses the best scheme from each pair?" We explore statistical techniques for fitting the model to the training data and use the receiver operating characteristic curve of signal detection theory to represent the power of the resulting models. The trained prediction methods predict whether fusion will beat an oracle, at levels much higher than could be achieved by chance.

Introduction

It is by now well established that the computational processes generally called Data Fusion can be quite effective in retrieval of text from large databases. This seems to be an extension of the principle observed some time ago by Saracevic and Kantor (1988), who considered the effects to be achieved by combining the search formulations developed by independent human search experts.

In the present work, we explore the potential for achieving similar results by combination of distinct retrieval schemes. We use the generic term "scheme" because often several different schemes can be realized using the same system of software, by varying the parameter settings that determine the ways in which documents are represented, and the algorithms used to estimate the relevance of documents to a given quest or search need (see Kantor et al., 1999, for a discussion of the notion of "quest," which extends the static concept of query or problem statement). Data Fusion (Varshney, 1997) can be accomplished at several different levels. Data Level Data Fusion combines distinct representations of the documents to be retrieved. Signal Level Data Fusion combines the signals (generally considered to be the Retrieval Status Value) generated by a particular system in the presence of a particular document. Decision Level Data Fusion deals with information produced after the system has completed its processing of the signal produced from the data, and has issued its report. In older systems, and in many engineering applications of signal detection, the report is a binary value—relevant, or not relevant. With modern retrieval systems the report can be a ranked list of documents in the order that the system judges will be most effective for resolution of the given quest.

In the present work we concentrate on Signal Level Data Fusion, and deal with the numerical scores assigned to documents by each of several different systems. These numerical scores, or retrieval status values, are used to generate the ranked list. However, because separations between scores will vary among pairs of adjacent items, the scores presumably contain more useful information than do the ranked lists. In this research we will study a procedure

Received 29 November 1999; Revised 2 May 2000; accepted 2 May 2000

that we call “Naive Data Fusion” (NDF). In the NDF process, each of the participating systems is considered to have “equal weight” in the data fusion process. Generally, to assign equal weight to each of several factors, one multiplies the signals or scores generated by those factors by a number that is inverse to some measure of the variation among those scores. In the construction of statistical indices, for example, a typical measure is the standard deviation of the scores. In the information retrieval setting, it seems more straightforward, and has been advocated in many previous works (Fox & Shaw, 1993, 1994) that the variation be represented by a range. Typically, this range is taken equal to the difference between the score of the highest ranked document and the score of the document at some fiducial position, such as the 100th or 1,000th document in the list. Because our experiments were conducted using the TREC data on the routing task (Harman, 1996), for which systems were encouraged to produce a list of 1,000 ranked documents, the range that we use is determined by the first and 1,000th retrieved documents.

Previous work on the predictive problem has concentrated on attempting to predict the performance of the fused system (Vogt & Cottrell, 1999). That work achieved impressively high levels of statistical success, as indicated by the *R*-squared coefficient. In the work presented here, we find statistically significant but substantially lower values for *R*-squared. The reason for this difference is that the effectiveness of a fused system will tend to be rather close to the effectiveness of the more powerful of the constituent schemes that are combined by data fusion. It will deviate by a relatively small amount above or below that scheme. In our tests, however, we are attempting to determine whether the fused schemes will do better than a hypothetical oracle. The oracle is presumed to be able to select the better of the two schemes for a given quest, in advance of running the retrievals. We have chosen the oracle as our baseline for comparison because of the apparently widely held belief that the most promising path for refinement of information retrieval techniques is the development of methods for matching schemes to quests, so that each specific quest is managed by the scheme that is “best suited to it.” Situations in which data fusion produces results better than those obtainable by an oracle highlight the ways in which data fusion methods transcend this widely held folk belief, and point the way to even greater achievements in retrieval.

There are many variables that deserve consideration in attempting to predict whether the fusion of two schemes will be more effective than the better of the two. These include the effectiveness of the two schemes themselves, which can be measured by any of a number of single indicators based on the precision–recall curve generated by the scheme for particular topics. These candidates are more concisely expressed through the introduction of the variable n, which represents the position of a document in the ranked retrieved list, and the cumulated number of relevant documents retrieved through that point in the list, represented by $g(n)$.

The recall at position n is defined as $g(n)/G$, where G is the total number of documents in the collection that are relevant to the present quest. The precision at position n is defined as $p_n = g(n)/n$. In the TREC setting (Harman, 1996), p_{100} is one of several published indicators of the effectiveness of the scheme. It is the most finely resolved indicator that does not depend on unproven assumptions about the overall nature of the problem. The other candidates, such as p_{1000}, involve the presumption that all documents not retrieved by any systems, somewhere in the top ranked 100, are definitely not relevant. This is not known to be true, and has informally been contested from year to year by participants in the TREC conferences. The same uncertainty surrounds the determination of G. For that reason we do not make use of the measure denoted p_{ave} (average precision vs. recall.) p_{ave} is a finer measure of system effectiveness, but depends on untestable assumptions. Thus, our single measure of system performance will be p_{100}.

Other naturally arising measures characterizing schemes to be entered into a data fusion involve the method or philosophy underlying the scheme. For example, some schemes represent text-only by single words; others use the same base but exclude frequently occurring stop words; others conflate various morphological forms of the same word by stemming; others extend the representational base to include pairs of consecutive words that occur frequently; others use more sophisticated methods for finding linguistically meaningful phrases in texts. In addition to these variations in representation, there are variations in the methods used for judging the similarity between a retrievable document and the current quest. In this arena there is a dazzling array of philosophical stances, supported by a somewhat narrower array of detailed computational methods. In particular, for the problem considered here, the routing task, the differences among the philosophies tend to be overpowered by the difficulties of the specific task.

In the routing task the participants in the TREC Conference are given a large set of previously judged documents for the same quest, and are permitted to use those to build the methods that they will use to handle retrieval from a new set of documents. In this situation most current schemes are forced to rely on some form of linear classifier, in which the score assigned to a document can be computed as the sum over a set of useful “terms,” of the product of three factors; a factor representing the importance of the term t in the “quest,” $q(t)$; a factor representing the importance of the term in the document, $d(t)$; and a factor representing the general usefulness of that particular term for the purpose of discriminating documents in the given collection, $m(t)$, often called the “Inverse document frequency” or (idf) factor.

In this situation we have decided not to attempt to classify schemes by the methods involved, but rather to develop an objective measure of scheme similarity that can be computed based only upon the results of applying the scheme to a body of standard quests. It is a measure based on the number of pairs of documents that are placed in different order by each of two IR schemes (see appendix).

When we consider the effectiveness of procedures for predicting the effectiveness of data fusion, we find that present methods for measuring the power of such a predictor procedure are less than satisfactory. Typical measures involve (a) developing a statistic that serves to predict the effectiveness of the data fusion, (b) selecting a cutoff value of that statistic, and (c) counting the number of correct classifications and misclassifications when this method is applied to a set of data prediction examples. We find this to be unsatisfactory because it does not take into account the fact that misclassifications of the two different kinds may bear different costs for the users of the system. In this setting the correct and complete tool to be used for evaluating the effectiveness of a prediction procedure is the "Receiver Operating Characteristic" (ROC) of the prediction method itself (Egan, 1975; Swets & Pickett, 1982). The ROC is a curve that represents the fraction of all effective cases that would be classified correctly, plotted as function of the fraction of all cases in which the fusion is ineffective, but is falsely predicted to be effective. In other words, it measures the ability of the predictive scheme to correctly *detect* effective data fusion, while keeping track of the degree to which the scheme generates "false alarms." The importance of the ROC lies in two facts: (1) it permits users of the procedure to select thresholds for classification that are appropriate to their own assessments of the relative cost of missing cases of effective fusion as opposed to false alarms in which they incorrectly predict effective fusion. (2) The relative strength of two procedures for predicting the effectiveness of data fusion can often be read easily from the ROC, because a prediction procedure whose ROC curve lies always above that of another procedure will be superior to it no matter what the user's specific estimates of costs and values may be.

The reminder of this article is organized as follows: in Section 2 we discuss our research question, the two predictive variables and the rule for simple symmetrical data fusion. In Sections 3 and 4, we analyze some of statistical relationship between the predictive variables and the effectiveness of data fusion. In Sections 5 and 6, we report our training of predictive models using 16,250 cases of data fusion and the results of applying such models to predict whether the outcome will be effective in another 11,385 cases of data fusion. Section 7 presents conclusion.

Research Question, Variables, Data and Fusion Rule

Explanatory Variables and Measures of Effectiveness

In a previous study (Ng & Kantor, 1998) we investigated two conditions for effective data fusion, which were: (1) the condition of similarity of efficacy, and (2) the condition of dissimilarity of outputs. In that study, we used the ratio of the precision at the 100th document (i.e., p_{100}) of two schemes to measure their similarity of efficacy, for example, if p_{100} of the poorer scheme is denoted by p_l, and of the better scheme is p_h, then the similarity is p_l / p_h. (From now on, we call this ratio r.) To measure the dissimilarity between two IR schemes, we used the number of out-of-order pairs between the two ranked output lists of two IR schemes. That is, if in one output list document A is ranked above document B while in the other output list B is ranked above A, this represents one out-of-order pair between the two output lists. We normalized this measure to have minimum value of 0 and maximum value of 1 so that it can be used for IR schemes with different cutoff points in their outputs. (For more details, see appendix; for rigorous definition and implementation algorithm of z, see Kantor, Ng, & Hull, 1998a; Ng, 1999; Ng & Kantor, 1998.) From now on, we call this variable z (normalized dissimilarity). Using r and z as predictive variables and 0.5 as prior probability for linear discriminant analysis, we achieved 73.0% correct classification (Ng & Kantor 1998).

In this study, we continue to investigate the predictive power of r and z. We train a model to predict whether a fusion scheme will perform better than an oracle who uses the best scheme for each pair. We use relative improvement in p_{100} compared to the better IR scheme to measure the effectiveness of data fusion. Let p_{100} of scheme S be p_{100} (S), let $S_1 \, f \, S_2$ represent a specific fusion of schemes S_1 and S_2, for a topic-wise comparison, we define the effectiveness of data fusion E ($S_1 \, f \, S_2$) as:

$$E(S_1fS_2) = \frac{p_{100}(S_1fS_2) - \max\{p_{100}(S_1),\, p_{100}(S_2)\}}{\max\{p_{100}(S_1),\, p_{100}(S_2)\}}$$

Data Sets

We use the output lists of the IR schemes produced for the routing tasks of the fourth and fifth Text REtrieval Conferences (i.e., TREC 4 and TREC 5, see Harman, 1996; Voorhees & Harman, 1997) as data for training and testing respectively. In the TREC 4 routing task, there were 26 schemes run on the full document collection for 50 topics, producing 16,250 cases of pairwise data fusion. In TREC 5 routing task, there were 23 schemes for 50 topics, however, five of the topics (topics 68, 125, 237, 240, and 243) had no judged relevant document in the collection. We only use the remaining 45 topics. We have 11,385 cases in our testing data set.

Method of Data Fusion

There are many successful data fusion experiments using the sum of normalized relevancy scores as the fusion rule (e.g., Belkin, Kantor, Fox & Shaw, 1995; Fox & Shaw, 1993, 1994). In our experiments, we use the same fusion rule for combination. The ordering of the fused list is determined by the sum of normalized relevance scores. The larger the sum of normalized relevancy score of a document, the higher its rank on the fused list. The normalized rele-

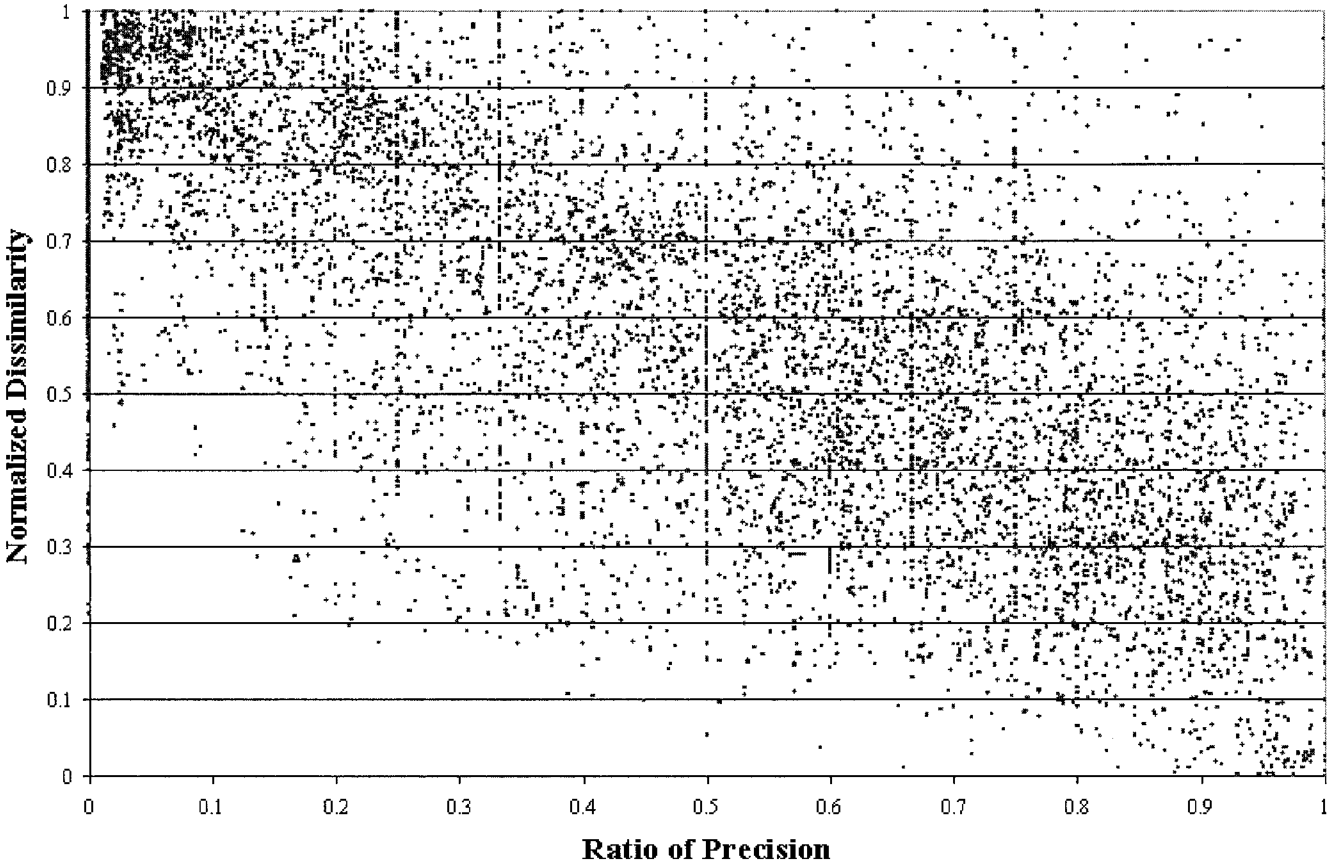

FIG. 1. Scatter plot of *r* versus *z* for negative cases of the training data set.

vancy score *s* of document d^j is calculated by the following formula:

$$s(d^j) = \frac{v(d^j) - v(d^{1000})}{v(d^l) - v(d^{1000})}$$

where *v(d)* is the score assigned to document *d*, and d^j is the document occupying position *j* in the ranked list under consideration. Such a transformation is applied to the scores generated for each quest considered, and separately for each scheme involved in the fusion process.

Exploratory Analysis

There are 16,250 cases of data fusion in our training data set. In 3,623 of these cases the performance (i.e., p_{100}) of the fused system is better than the best of the two original schemes. For convenience, we will refer to these as the positive cases. There are 9,171 cases with performance worse than the best of the two original schemes (i.e., negative cases). To understand how these two outcomes are related to the explanatory variables considered, we plot two scatter plot graphs of *z* (normalized dissimilarity) versus *r* (ratio of p_{100}) for these 9,171 and 3,623 cases (Figa. 1 and 2).

Figure 1 is the scatter plot of *r* versus *z* for the negative cases. We can see that there are very few cases in the region of small *r* and small *z*. For example, there is no case in the region [$r < 0.2$, $z < 0.2$], and there are only about 10 cases in the region [$r < 0.4$, $z < 0.4$]. There are also comparatively very few cases in the region of high *r* and high *z*. It appears that the negative cases tend to scatter generally around the line defined by the equation $z + r = 1$. As *r* approaches 1, *z* approaches 0; and as *r* approaches 0, *z* approaches 1. This means that for the negative cases, when the performances (i.e., p_{100}) of the two IR schemes are more or less the same, their output lists are similar to each other.

In Figure 2, we apply the same analysis to the positive cases. We see that there are very few positive cases in the region of small *r* and *z*. There is no case in the region [$r < 0.2$, $z < 0.2$], and there are only two cases in the region [$r < 0.4$, $z < 0.4$]. There are comparatively fewer cases in the region of large *r* and *z*. However, in contrast to the negative cases, the positive case are more likely to lie above the diagonal $r + z = 1$, and are more heavily concentrated at the right side of the scatter plot, near the line $z = 1$. This indicates that schemes with *dissimilar* outputs but *comparable performance* are more likely to give rise to effective naive data fusion.

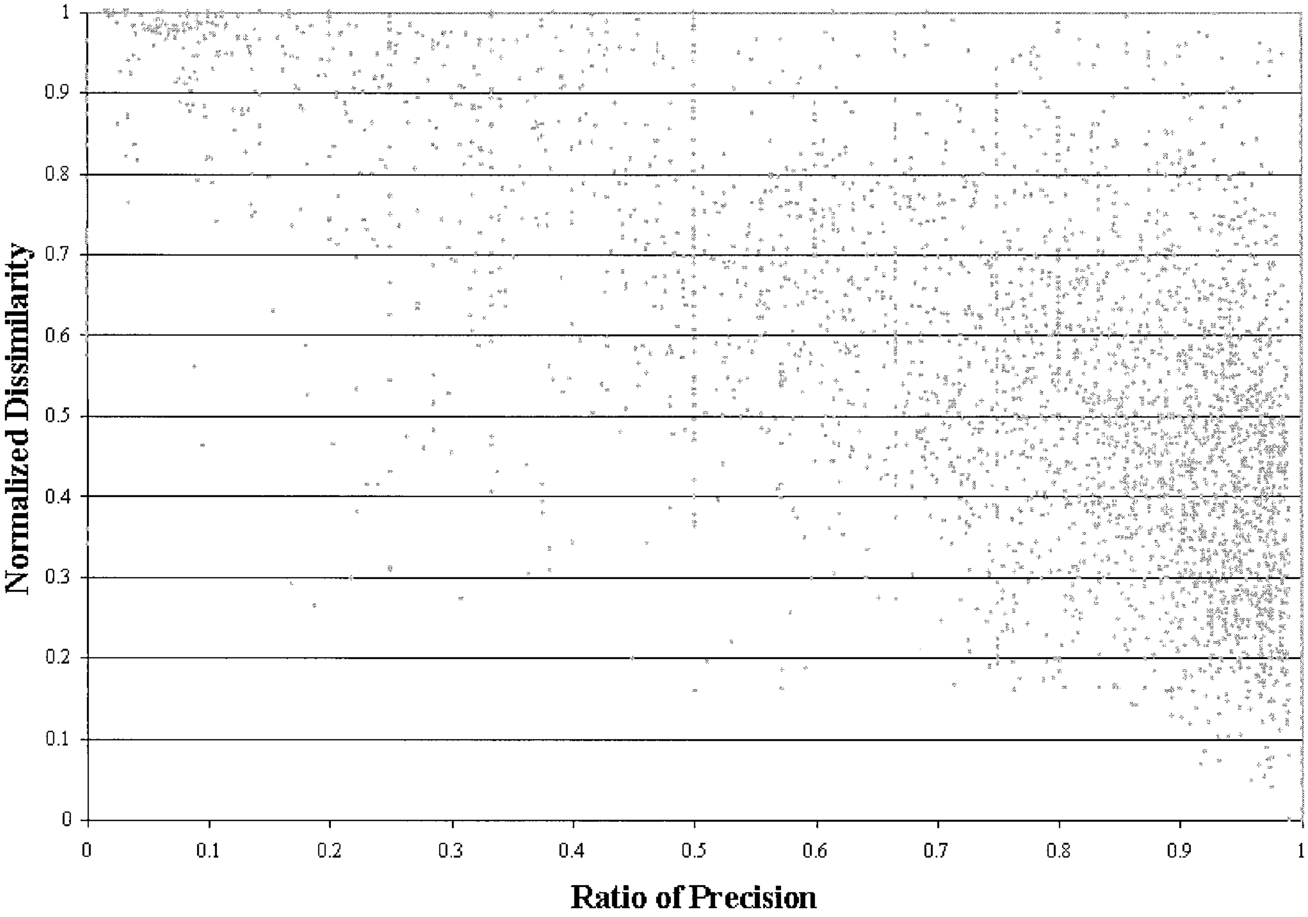

FIG. 2. Scatter plot of *r* versus *z* for positive cases of the training data set.

Parametric Statistical Methods

To convert the exploratory observations presented above into concrete procedures for predicting whether the fused schemes will be more effective than an oracle, we have applied three statistical techniques to the training data: linear discriminant analysis, multiple linear regression analysis, and logistic regression analysis. Our goal was to determine which techniques are most effective in assigning a classification score to the training data set. The classification scores will be used to generate ROC curves. Every point on the curve represents a possible cutoff point for separating positive and negative cases. The coordinates in the ROC plot are the corresponding detection rate (vertical axis) and false alarm rate (horizontal axis). The closer an ROC curve comes to the ideal point (100% detection, 0% false alarm rate), the better the performance of the predictor. As the threshold for predicting whether naive data fusion will be effective is varied smoothly, the curve rises in a concave fashion, and the detection rate is always higher than the false alarm rate. If one statistical technique produces an ROC with higher detection rate at any false alarm rate, it is preferred in predicting the effectiveness of data fusion on the testing data set. The resulting behaviors, as exhibited in the ROC curves on both training and testing data, are essentially the same for the three analyses. Therefore, only the logistic regression analysis will be described in detail.

Linear Discriminant Analysis

We seek a linear combination of the variables *r* and *z* that will discriminate between the positive and negative groups in the training data set in such a way that the ratio of between-group sum of squares to the within-group sum of squares is maximum. We drop those cases with no improvement ($E = 0$) from the analysis, leaving 12,794 cases. Using the SPSS package we find that the discriminant variable is $d = 3.195\ z + 4.643\ r - 4.36$. The sign of *d* is chosen so that larger values of this variable scale are more likely to correspond to cases of effective data fusion. We plot an ROC curve by sorting the cases in decreasing order of the discriminant scores. As we mentioned in the above paragraph, the ROC curve is essentially the same as the one based on logistic regression analysis ,and we will discuss it in more detail later.

Multiple Linear Regression Analysis

In the linear discriminant analysis we did not consider the magnitude of the effectiveness of data fusion, and used

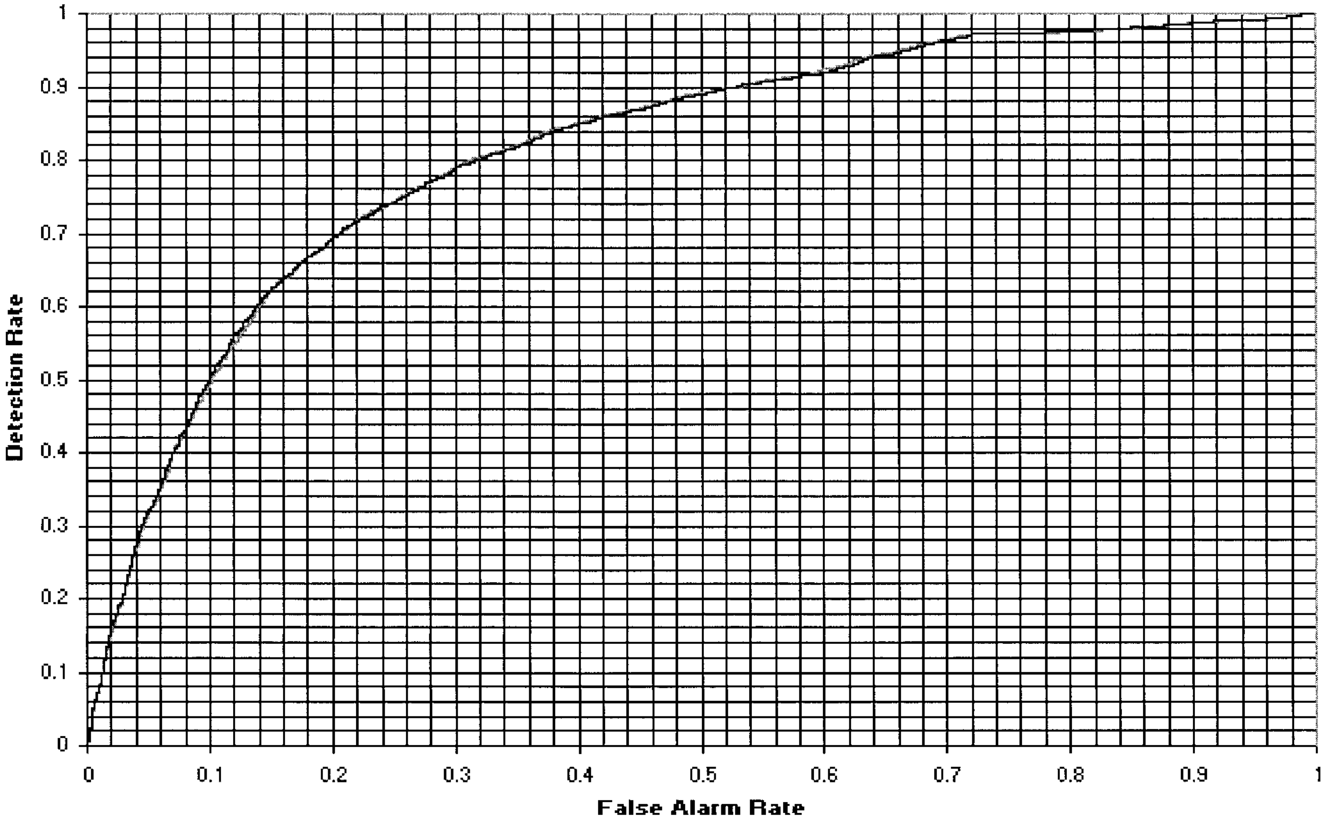

FIG. 3. The ROC curves of discriminant analysis and logistic regression overlap with each other and are almost visually indistinguishable.

only its sign in our analysis. In this section, we ask whether the unused information, combined with multiple linear regression, will give us a better ROC curve. In applying multiple linear regression analysis we test whether the dependent variable, *E* (effectiveness of data fusion), is linearly related to the two independent variables, and then calculate the strength of the linear relationship. We can then use the sign of the predicted effectiveness (*E*) to classify all the cases, and generate an ROC curve. In the 16,250 cases of training data, 495 cases have $p_{100} = 0$ for both IR schemes, so *E* cannot be defined. We eliminate those cases from analysis, using only the remaining 15,755 cases. The regression equation estimated by the OLS method is: $E = 0.408\ r + 0.164\ z - 0.418$. The R^2 is 0.204. To produce the ROC curve we sort cases in descending order of the predicted value of *E*. At each point in the sorted list, we calculate the detection and false alarm rate using that point as the cutoff for prediction. The resulting ROC curve is more or less the same as the discriminant analysis ROC curve. Generally the difference between the two ROC curves is about 0–2% along the vertical axis (detection rate), and it never exceeds 2%. Thus, when applied to the training data, it seems that the classificatory power of the two methods is similar.

Logistic Regression Analysis

In the previous section, we compared multiple regression analysis and discriminant analysis on the training data set. While multiple regression analysis has used more data (15,755 cases) and more information about the dependent variable (not just the sign but also the magnitude of the effectiveness of data fusion), it does not offer a generally better ROC curve, but more or less the same operating characteristic as discriminant analysis. Thus, it seems reasonable to use the discriminant function on the test data set to see how well it can predict cases of effective data fusion. However, there are some inherent limitations associated with discriminant analysis, which lead us to consider one more method, logistic regression analysis.

In linear discriminant analysis, two assumptions must be met for the prediction rule to be optimal. The first is the assumption of multivariate normality of independent variables. The second is the assumption of equal variance–covariance matrices in the two groups. The logistic regression model requires far fewer assumptions than does discriminant analysis; and even when the assumptions required for discriminant analysis are satisfied, logistic regression still performs well (Hosmer & Lemeshow, 1989). Here we use logistic regression to investigate the discriminating power of the two independent variables. As with the linear discriminant analysis, we eliminate from the training set those cases where the effectiveness is zero.

Let Pr(E>0) represents the probability that the simple symmetrical data fusion between the output lists of two IR schemes will be better than the best of S_1 and S_2 (i.e., positive data fusion effectiveness). Let Odds(E>0) denote

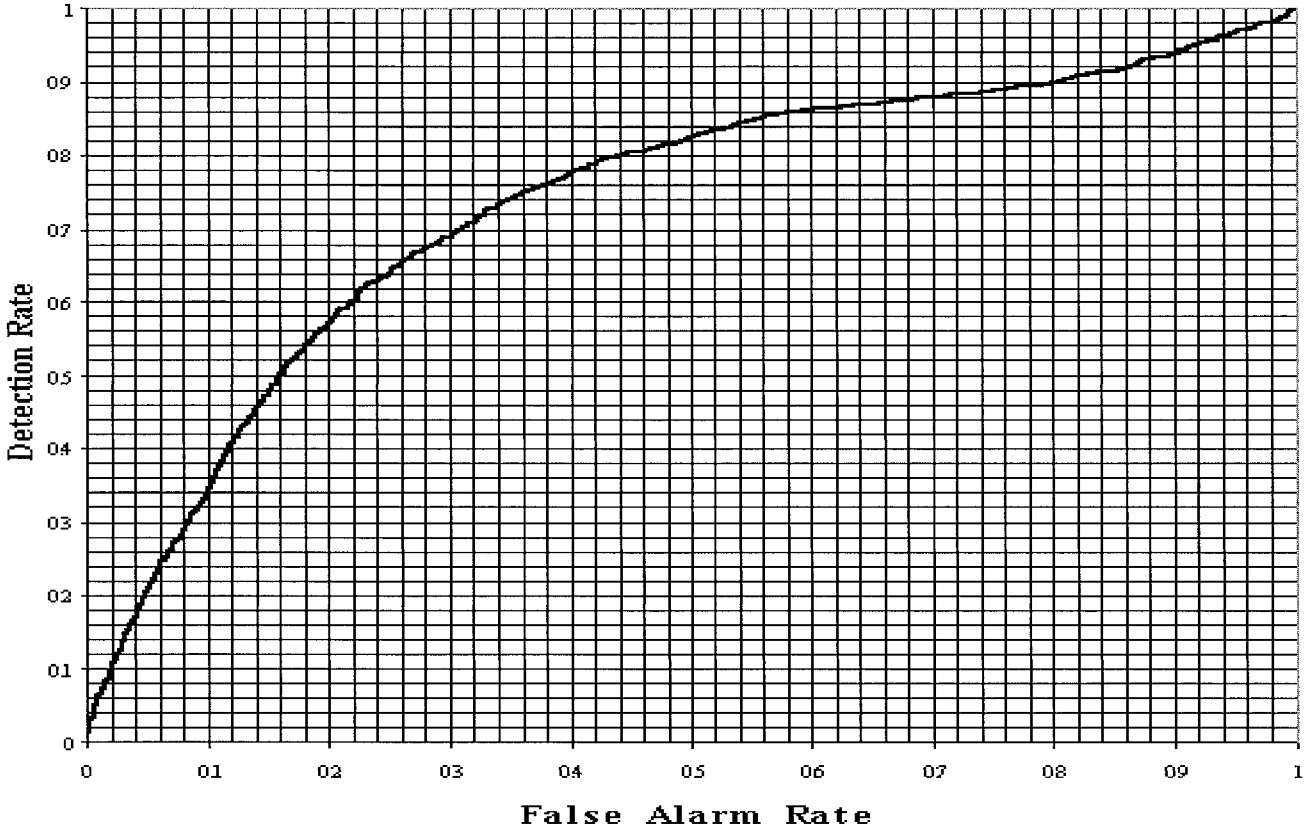

FIG. 4. The ROC curve of predicted probabilities by logistic regression.

the odds of positive data fusion effectiveness (that is, the ratio of Pr($E>0$) to 1 − Pr($E>0$). Using the natural logarithm of Odds as dependent variable, the equation for the relationship between the dependent and independent variables becomes: $\log_e$ (Odds ($E>0$)) = $B_0 + B_1\, r + B_2\, z$. Converting the odds back to the Pr($E>0$), we have Pr($E>0$) = Odds($E>0$) /[1 + Odds($E>0$)]. This yields a model function estimating the probability that data fusion is effective. We use the training data to estimate the coefficients in the above equation, B_0, B_1, and B_2.

In multiple linear regression analysis, the parameters of the model were estimated by the OLS (ordinary least square) method. OLS method selects regression coefficients that result in the smallest sums of squared distance between the observed effectiveness of data fusion and the predicted effectiveness of data fusion. In logistic regression analysis, the parameters of the model are estimated using the MLE method (maximum-likelihood method). For the MLE method, the coefficients that make the observed effectiveness of data fusion most "likely" are selected (i.e., those that maximize the probability of the observed results, given the parameter estimates.) After four iterations, the change of log likelihood decreases by less than 0.01% and the estimation was terminated. The resulting logistic regression equation is Odds($E>0$) = $e^{6.4485 - 5.6068r - 3.6956z}$. If we classify all cases with Pr($E>0$) > 0.5 as positive cases, the overall classification accuracy for the logistic regression is 78.57%.

With the logistic regression equation, we can order cases by the estimate of the probability that data fusion will be effective, and plot an ROC curve. The ROC curve of logistic regression is almost identical to the ROC curve of linear discriminant analysis. If we put the two ROC curves in one figure, they are almost visually indistinguishable (Fig. 3).

We see in Figure 3 that a model achieves a detection rate of about 60%, with a false alarm rate of only about 14%. In other words, with this threshold setting it would correctly label 60% of the cases showing improvement, and would incorrectly label only 14% of the cases showing no improvement. Inspecting the original data that generated the ROC curves, we can see that, with the same false alarm rate, sometimes the detection rate of logistic regression is better than that of discriminant analysis, sometimes vice versa. In average, the difference in detection rate is less than 0.2%.

The Predictive Power of Logistic Regression

The ROC curves of the three methods we used are very similar to each other. In fact, they are also essentially the same when applied to the testing data. Therefore, in the following, we will only report the result of applying logistic regression.

We apply the logistic regression equation estimated from the training data set to compute the probability of effective data fusion for the testing cases, and use the predicted

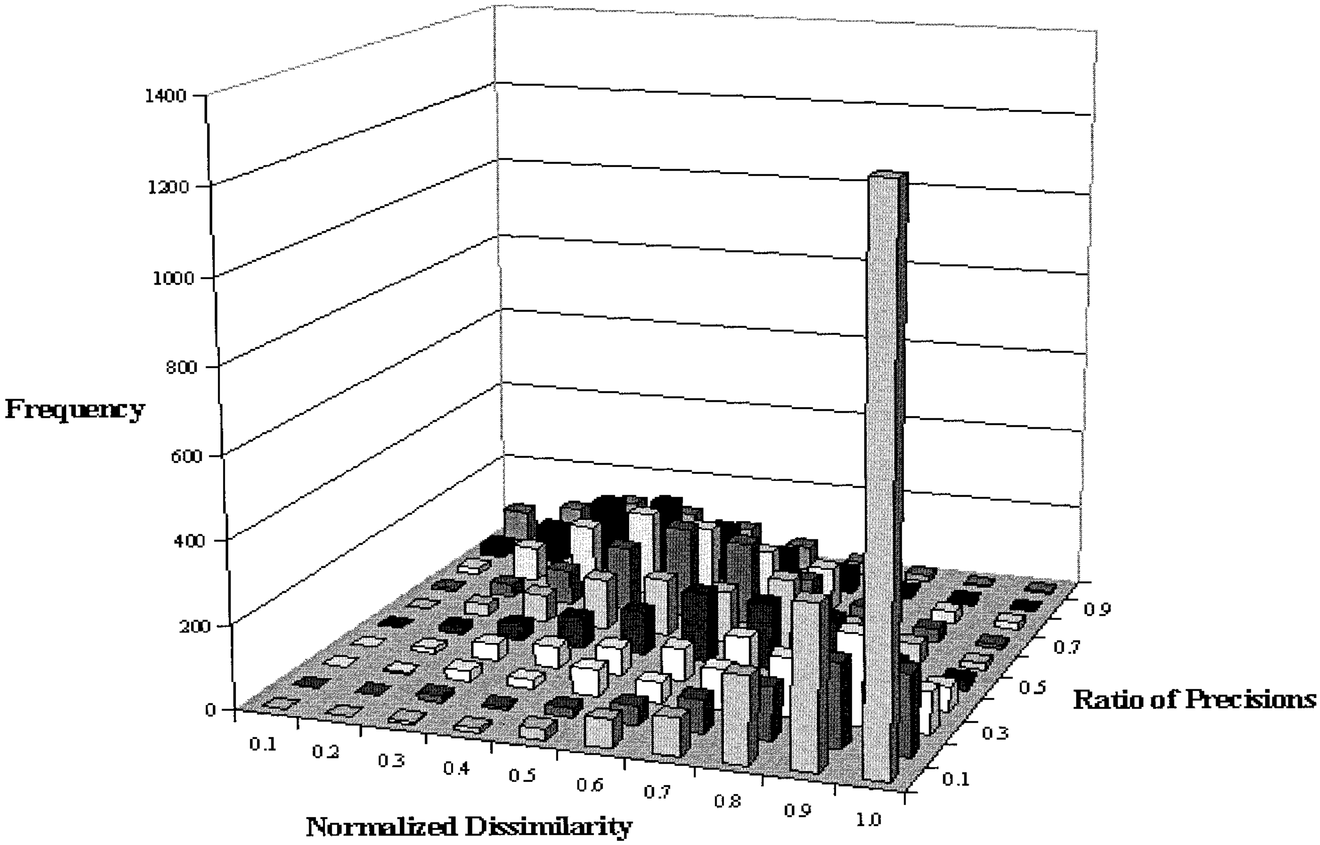

FIG. 5. Frequency distribution of negative cases of the training data set by z and r.

probability to plot an ROC curve (Fig. 4). From Figure 4, we can see that when the detection rate is below about 75%, the predictive power is much better than (random) prediction without information about normalized dissimilarity and precision ratio. Below about 75%, the detection rate is one to two times higher than the false alarm rate. When the detection rate is higher than 75%, it is not much higher than the false alarm rate, but still better than chance alone. For example, when the detection rate is about 80%, the false alarm rate is about 47%. To achieve a detection rate of 90%, we would have to tolerate a false alarm rate of about 80%.

Nonparametric Training and Testing

The above techniques are parametric techniques. For comparison, we apply a nonparametric estimation of the ratio of two distributions, which is totally empirical and highly nonlinear. We will call this the "Bin-Ranking method."

We "bin" the training cases into 100 (square) regions by splitting r and z into 10 equally spaced bins. Comparing the frequency distributions for the positive and the negative cases in these bins gives us a more precise empirical picture of how variation in r and z is related to the occurrence of effective data fusion. By ordering the 100 bins by decreasing ratio of effective to ineffective data fusion, we have a prediction for new, test, data. Figure 5 shows the frequency of the negative cases in the training data as a function of r and z. The cases are grouped into 100 small regions. The height of the bar in each region represents the number of cases occurring in that region. From Figure 5, one can also see that the cases scatter more or less evenly around the negative diagonal. In addition, there are more than 1,000 cases located in the region $(0 \le r < 0.1, 0.9 \le z < 1)$, while the other 99 regions have lower frequencies. These 1,000 plus cases are from combinations of IR schemes very dissimilar in terms of both performance and outputs.

This observation does not imply that dissimilar IR schemes are more likely to yield ineffective data fusion. It depends on the ratio of positive to negative cases in the same region, not on the ratio of the number of negative cases in that region to that of negative cases in other regions. So we examine the distribution of positive cases (Fig. 6) grouped into the same 100 bins.

In sharp contrast to the negative cases, the highest frequency bars of the positive cases are not located in the lowest ratio of precisions regions $(r < 0.1)$ but in the highest ratio of precision regions $(r > 0.9)$. This observation reflects the discriminant power of r. In addition, the highest bars are not in the region of highest dissimilarity but in the region of medium dissimilarity.

We calculate a ratio by dividing the number of positive cases in a bin by the number of negative cases in the corresponding bin. We rank the bins, with the cell having

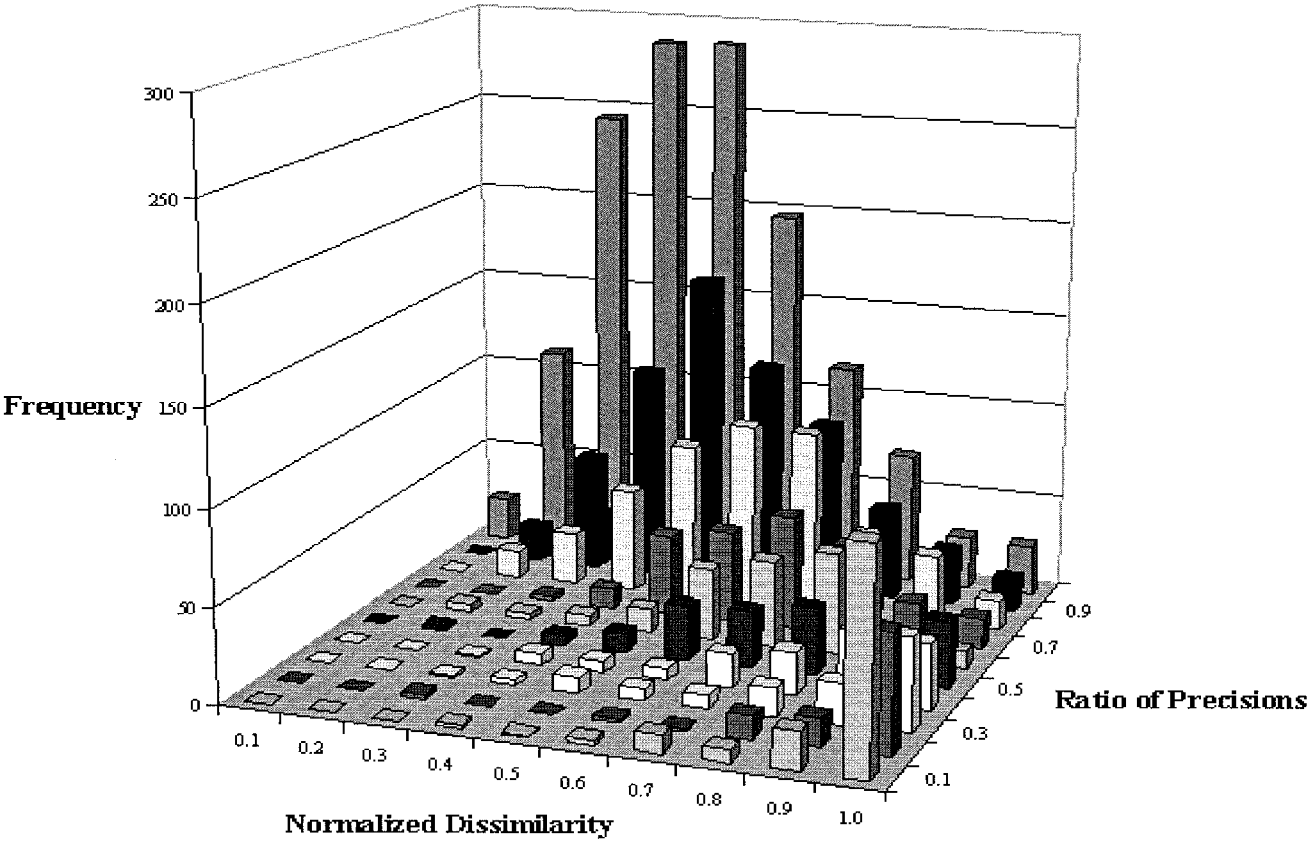

FIG. 6. Frequency distribution of positive cases of the training data set by *z* and *r*.

the highest ratio ranked as "1." The results are shown in Table 1. The lowest rank (shared by many cells) is 83.

The ROC curve produced has 83 data points. Point *i* represents the detection rate and false alarm rate if we use rank *i* as the cutoff point for declaring effective fusion. The detection rate at point *i* is the total number of positive cases in the bins at ranks 1 through *i*, divided by the total number of positive cases. The false alarm rate of point *i* is the total number of negative cases in the bins at ranks 1 through *i*, divided by the total number of negative cases.

Figure 7 is the ROC curves of this method and logistic regression. We see that, on the training data, the ROC curve of this nonparametric method is more powerful than that of the parametric method. With the same false alarm rate, the detection rate of the nonparametric method is always higher than that of the logistic regression. However, any empirical fitting of distribution curves, and of their ratios, runs the risk of overfitting the data. To test the relation observed in Figure 7, we rank the bins of the testing data set according the ranks determined by the training set. Repeating the preceding analysis we can compute the ROC curve of this empirical prediction, for the test data (Fig. 8).

From Figure 8, we see that the ROC curve for bins of test data, arranged in the order determined by the training data

TABLE 1. Each cell contains the rank order of the relative performance of a range of *r* and *z*.

		Normalized dissimilarity									
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Ratio of precisions	0.1	83	83	83	44	78	74	58	80	72	66
	0.2	83	83	55	83	83	76	82	63	67	35
	0.3	83	83	77	52	54	56	69	60	61	30
	0.4	83	83	81	57	65	68	53	51	37	29
	0.5	83	59	83	70	62	49	46	41	28	15
	0.6	83	50	75	73	64	36	40	23	34	26
	0.7	83	83	79	71	45	42	31	24	25	14
	0.8	83	48	47	39	32	22	20	19	17	21
	0.9	83	43	33	27	16	13	12	10	8	7
	1.0	38	18	11	9	6	5	4	3	2	1

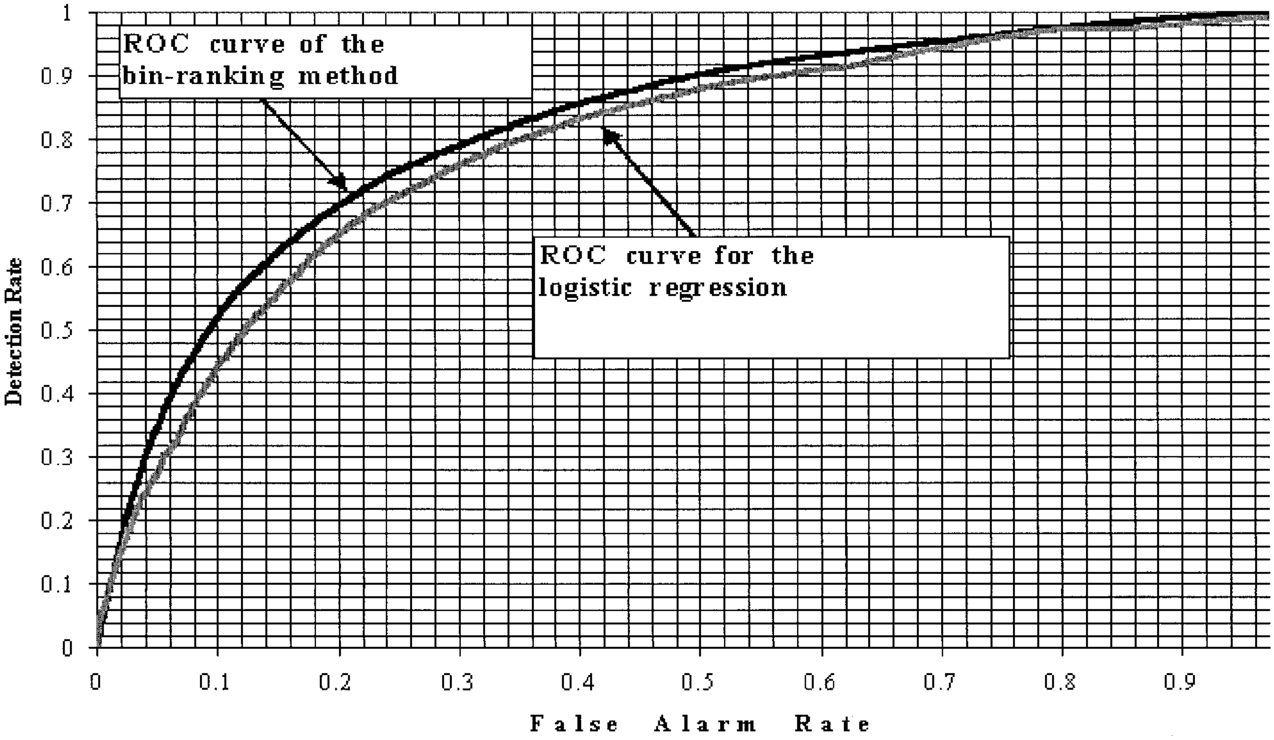

FIG. 7. ROC curves of the bin-ranking method and the logistic regression for the training data set.

is not always concave. An example is the point where the detection rate is about 60% and the false alarm rate is about 21%. These turning points indicate that the bins corresponding to the adjacent points are not in the best rank order.

We find that although the bin ranking method is more powerful than the parametric methods in classifying the training cases, its predictive power drops considerably when applied to the testing data set. Of course it is unlikely that

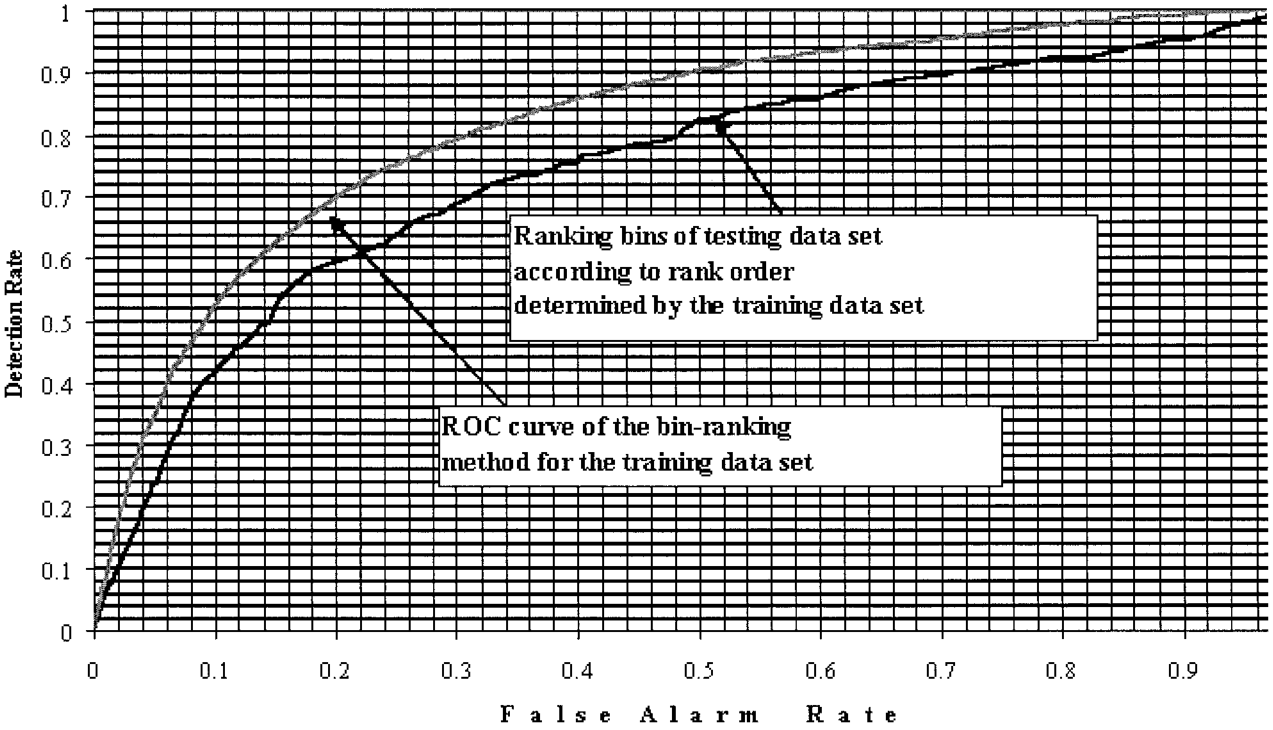

FIG. 8. The ROC curves of the bin-ranking method: training and testing.

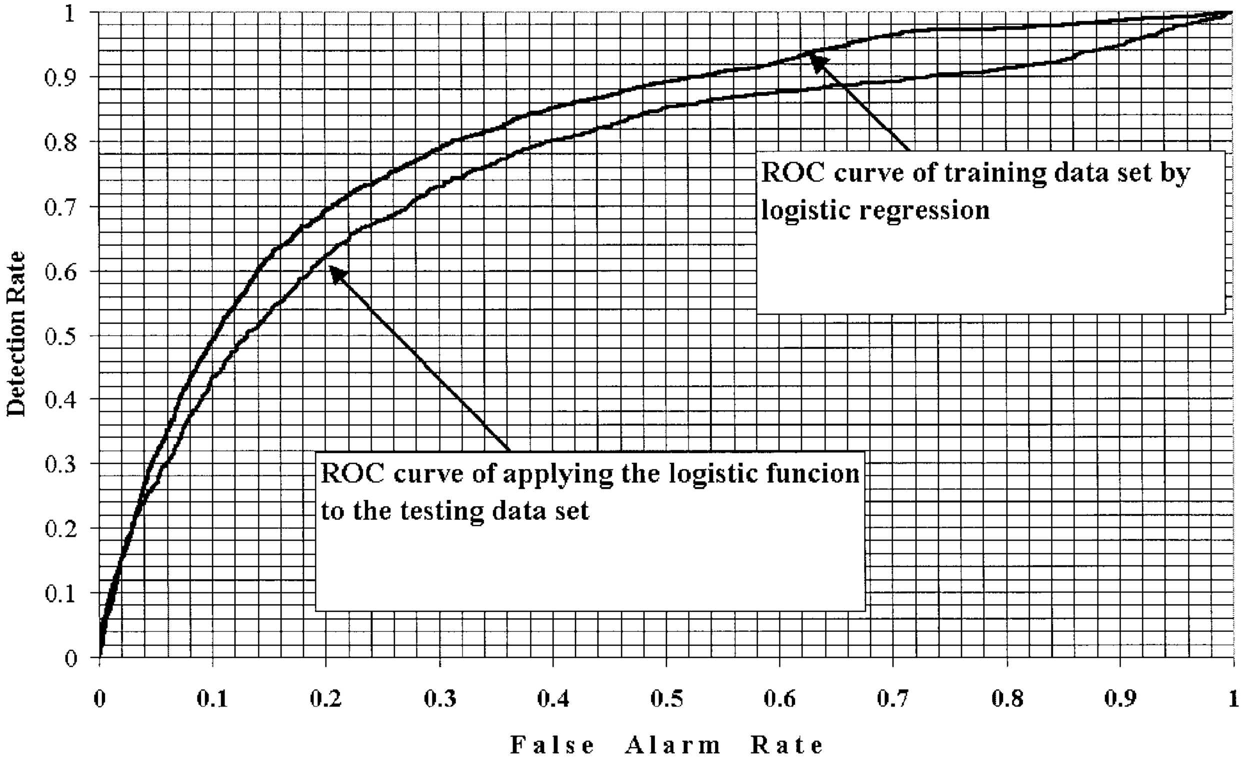

FIG. 9. The performance of logistic regression on the training data set and the performance of the same function applied to the testing data set.

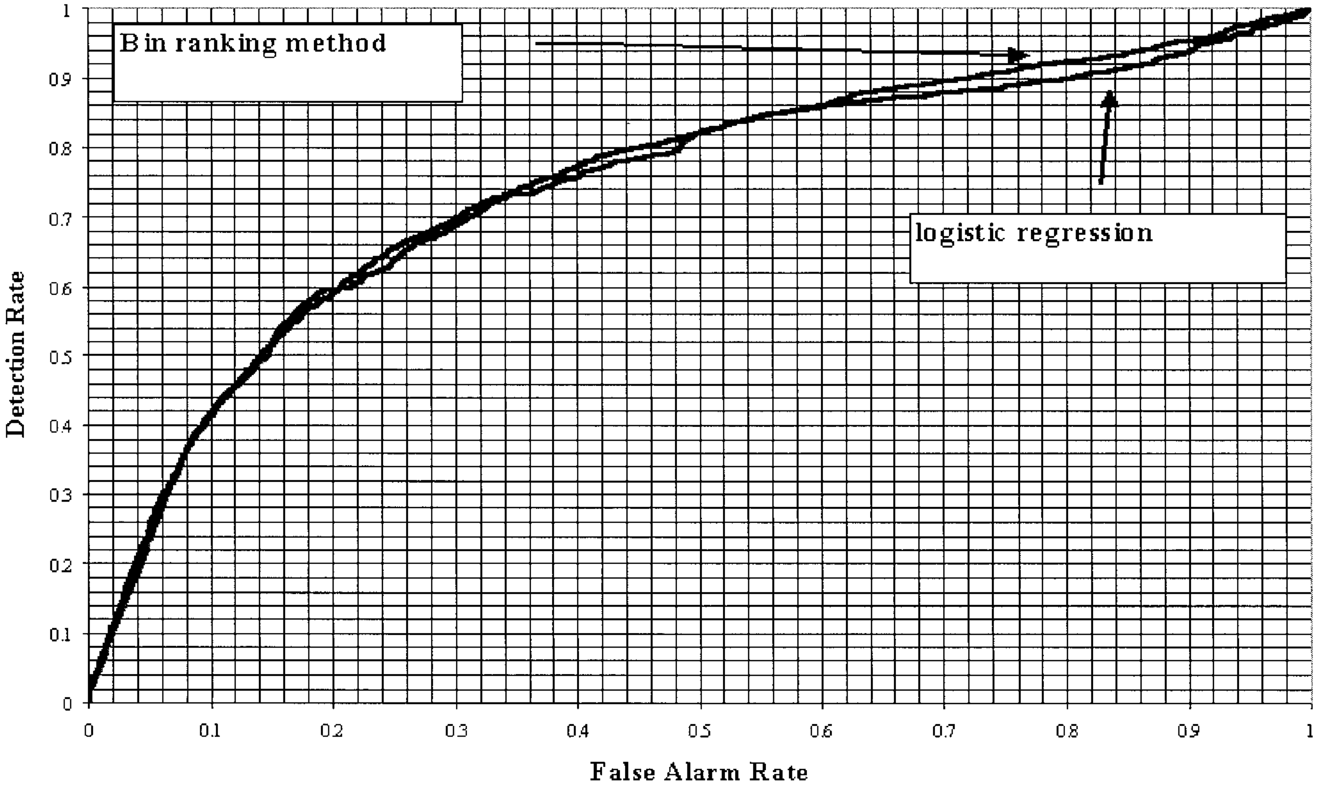

FIG. 10. The prediction performance of the two methods in the testing data set: logistic regression and nonparametric bin ranking method.

TABLE 2. Performance of different predicting methods.

Predicting method		Detection rate	False alarm rate
Logistic regression	Training	76%	24%
	Testing	69%	31%
Non-parametric bin ranking	Training	75%	25%
	Testing	69.5%	30.5%

applying parameters estimated from training data to testing data will give the same performance. However, the difference between the training stage and testing stage seems to be quite large. The ROC curves for logistic regression does not change so much from the training data set to the testing data set (Fig. 9).

Figure 10 is the ROC curves of the two predictive methods: parametric logistic regression and the nonparametric bin ranking method. When the false alarm rate is below about 60%, the two ROC curves are very close to each other, with less than 1% difference in detection rate, indicating that the predictive power of the parametric method and the nonparametric method are more or less the same. However, when the false alarm rate is above 60%, the ROC curve of the nonparametric method is always better than the parametric method by about 2%.

Conclusion

The two methods are quite comparable, in the sense that they have very similar ROC curves when apply to the testing data set. Comparatively speaking, the predictive power of the nonparametric bin ranking method has the highest detection rate for the region of false alarm rate greater than approximately 60%. In the region where false alarm rate is less than approximately 60%, the ROC curves of both methods essentially overlap with each other.

If we pick the point on the ROC curve where detection rate + false alarm rate = 1 as a convenient point for comparison, the performance of the two methods is the same. The results are summarized in the following table (Table 2).

The two predictive variables do not completely determine whether simple (linear) and symmetrical data fusion will be effective. We used ratio of precision at the 100th document to represent efficacy similarity and develop a new measure to represent scheme dissimilarity. A challenging direction for further studies is to explore other implementation of these concepts. For examples, using average precision, or other performance measure, instead of precision at the 100th document, which may be a more accurate representation of efficacy, may produce better result. Also, in our implementation of the scheme dissimilarity, we assign 0.5 to those pairs for which both documents only appear in one output list but not in the other. These may generate a lot of noise if we have many such documents. There may be another implementation that may be better (e.g., moving up the cut off points—from the 1,000th document, i.e., the end of an output list—in calculating z may decrease the number of such documents.)

This research is focused on predicting the sign of the effectiveness of data fusion. Another direction for research is moving on from predicting the sign of effectiveness to predicting magnitude of the effectiveness.

Acknowledments

This work was supported in part by the Data Management Program of the National Science Foundation under grant 9812086.

Appendix: Definition of Inter-IR Scheme Dissimilarity

To quantify the dissimilarity between two IR schemes, we need an objective measure. Because our concern is the operational performance, not the underlying mechanism or algorithm, instead of defining the distance between IR schemes, we measure the distance between their outputs. Mathematically, a ranked list of N items can be broken down into $1/2N(N-1)$ ordered pairs. From such a set of ordered pairs, only one ranked list can be reconstructed. For example, for a ranked list with three items A, B, C such that $A>B>C$ (We use $>$ to represent hierarchical relationship in the ranked list), the list generates $1/2 \times 3 \times (3-1) = 3$ ranked pairs, i.e., $A > B$, $A > C$, $B > C$. From these three ranked pairs, we can reconstruct exactly one ranked list which contains all the three items, i.e., the original ranked list, $A > B > C$. In other words, the ranked list and the set of ranked pairs are alternative ways of representing the ordering among the elements. Therefore, when comparing two ranked lists, instead of comparing two ranked lists directly, we can compare the ordered pairs determined by each list.

If we have two ranked lists containing the same elements, we can break down these two lists into two sets of ranked pairs. For example, for two ranked lists: $A > B > C$ and $B > A > C$, each generates three ranked pairs. For the first list, they are, $A > B$, $A > C$, $B > C$. For the second list, they are $B > A$, $A > C$, $B > C$. The first of the three pairs is out of order with respect to each other ($A>B$ vs. $B>A$), while the second and third pairs are in the same order. This representation of ranked lists can be used to calculate the distance between two ranked lists. We call the count of out of order pairs the non-normalized distance between the two lists.

In an IR ranked output environment, we can use the number of out-of-order pairs to measure the dissimilarity between the ranked outputs of two IR schemes. When the collection is huge, it is not likely for an IR system to offer the user a full output list. Therefore, we will only compare the top portion of the ranked output lists, not full lists, of different IR schemes. In this case, it is likely that the documents in the two lists are not exactly the same. When the two ranked lists have different elements, we may encounter two new situations: (1) for a given pair in one list, only one element is present in the other list; (2) for a given pair in one list, neither element is in the other list. For a pair from one list such that only one of the documents is in the other list, logically the order of the pair in the other list can be easily determined because the missing document must be in the lower part of the other list, below the cutting point. For those pairs of which both documents only appear in one list but not in the other, they can be either out-of-order or not, with equal probability (because there are as many permutations of the list in which the pair is in order as there are with it out of order.) Therefore, we treat the out-of-order scores for those pairs as "0.5." After we have the total out-of-order score, we can normalize it to be between 0 and 1. For more details and the implementation algorithm, see Ng and Kantor (1998), Kantor et al. (1998a, 1998b), and Ng (1999).

References

Belkin, N.J., Kantor, P.B., Fox, E., & Shaw, J. (1995). Combining the evidence of multiple query representations for information retrieval. Information Processing and Management, 31(3), 431–448.

Egan, J.P. (1975). Signal detection theory and ROC analysis. New York: Academic Press.

Fox, E.A., & Shaw, J.A. (1993). Combination of multiple searches. In Proceedings of the second text retrieval conference (TREC-2). National Institute of Standards and Technology Special Publication 500-215.

Fox, E.A., & Shaw, J.A. (1994). Combination of multiple searches. In Proceedings of the third text retrieval conference (TREC-3). National Institute of Standards and Technology Special Publication 500-215.

Harman, D. (1996). Overview of the fourth text retrieval conference. In D. Harman (Ed.), Proceedings of the fourth text retrieval conference. Washington, DC: GPO.

Hosmer, D.W., & Lemeshow, S. (1989). Applied logistic regression. New York: John Wiley & Sons.

Kantor, P.B. (1995). Tutorial on data fusion in information retrieval. Seattle, WA: ACM SIGIR.

Kantor, P.B., Hull, D., & Ng, K.B. (1998a). Advanced approaches to the statistical analysis of TREC information retrieval experiments. Technical report submitted to National Institute of Standards and Technology.

Kantor, P.B., Ng, K.B., & Hull, D. (1998b). Comparison of system using pairs-out-of-order. Technical report submitted to submitted to National Institute of Standards and Technology.

Kantor, P.B., Melamed, B., Boros, E., & Menkov, V. (1999). The information quest: A dynamic model of user's information needs. In Woods, Larry (Ed.) Proceedings of the 62nd Annual Meeting of the American Society for Information Science, 36, 536–545.

Ng, K.B. (1999). An investigation of the conditions for effective data fusion in information retrieval. School of Communication, Information and Library Studies. Ph.D. Thesis.

Ng, K.B., & Kantor, P.B. (1996). Two experiments on retrieval with corrupted data and clean queries in TREC 4 adhoc task environment: Data fusion and pattern scanning. In D. Harman (Ed.), Proceedings of the fourth text retrieval conference. Washington. DC: GPO.

Ng, K.B., & Kantor, P.B. (1998). An investigation of the conditions for effective data fusion in information retrieval: A pilot study. Proceedings of the 61th Annual Meeting of the American Society for Information Science (pp. 166–178).

Saracevic, T., & Kantor, P. (1988). A study of information seeking and retrieving. III. Searchers, searches, overlap. Journal of the American Society for Information Science, 39(3), 197–216.

Swets, J.A., & Pickett, R.M. (1982). Evaluation of diagnostic systems: Methods from signal detection theory. New York: Academic Press.

Varshney, P.K. (1997). Scanning the issue: Special issue on data fusion. Proceedings of the IEEE, 85(1), 3–5.

Vogt, C.C. & Cottrell, G.W. (1999). Fusion via a linear combination of scores. Information Retrieval, 1(3).

Voorhees, E.M., & Harman, D. (1997). Overview of the fifth text retrieval conference. In D. Harman (Ed.), Proceedings of the fifth text retrieval conference. Washington, DC: GPO.