Efficient Feature Selection in the Presence of Outliers and Noises*

Shuang-Hong Yang and Bao-Gang Hu

National Lab of Pattern Recognition(NLPR) & Sino-French IT Lab(LIAMA)
Institute of Automation, Chinese Academy of Sciences
{shyang,hubg}@nlpr.ia.ac.cn

Abstract. Although regarded as one of the most successful algorithm to identify predictive features, *Relief* is quite vulnerable to outliers and noisy features. The recently proposed *I-Relief* algorithm addresses such deficiencies by using an iterative optimization scheme. Effective as it is, I-Relief is rather time-consuming. This paper presents an efficient alternative that significantly enhances the ability of Relief to handle outliers and strongly redundant noisy features. Our method can achieve comparable performance as I-Relief and has a close-form solution, hence requires much less running time. Results on benchmark information retrieval tasks confirm the effectiveness and efficiency of the proposed method.

1 Introduction

Feature subset selection is a process of identifying a small subset of highly predictive features out of a large set of candidate features which might be strongly irrelevant and redundant [2,3]. It plays a fundamental role in data mining, information retrieval, and more generally machine learning tasks for a variety of reasons [3]. In the literature, many feature selection methods approach the task as a search problem [3,4], where each state in the search space is a possible feature subset. Feature weighting simplify this problem by assigning to each feature a real valued number to indicate its usefulness, making possible to select a subset of features efficiently by searching in a continuous space rather than a discrete state space.

Among the existing feature weighting methods, *Relief* [5,7,9] is considered one of the most successful ones due to its effectiveness, simplicity and efficiency. Suppose we are given a set of input vectors $\{\mathbf{x}_n\}_{n=1}^N$ along with corresponding targets $\{y_n\}_{n=1}^N$, where $\mathbf{x}_n \in \mathbf{X} \subset \mathcal{R}^D$ is a training instance (e.g., the *vector space model* of a document) and $y_n \in \mathbf{Y}=\{0,1,\ldots,C\text{-}1\}$ is its label (e.g., the category of the document), N, D, C denote the training set size, the input space dimensionality and the total number of categories respectively. The d-th feature of $\mathbf{x}$ is denoted as $x^{(d)}$, d=1,2,...,D. Relief ranks the features according to the weights w_d's obtained from a convex optimization problem [9]:

* This work is supported in part by NSFC (#60073007, #60121302).

H. Li et al. (Eds.): AIRS 2008, LNCS 4993, pp. 184–191, 2008.

$$\begin{aligned} &\mathbf{w} = \arg\max \sum_{n=1}^{N} \mathbf{w}^T \mathbf{m}_n \\ &s.t. : ||\mathbf{w}|| = 1, w_d \geq 0, d = 1, 2, ..., D \end{aligned} \tag{1}$$

where $\mathbf{w}=(w_1, w_2, ..., w_D)^T$, $\mathbf{m}_n = |\mathbf{x}_n - M(\mathbf{x}_n)| - |\mathbf{x}_n - H(\mathbf{x}_n)|$ is called the margin for the pattern $\mathbf{x}_n$, $H(\mathbf{x}_n)$ and $M(\mathbf{x}_n)$ denote the nearest-hit (the nearest neighbor from the same class) and nearest-miss (the nearest neighbor form different class) of $\mathbf{x}_n$ respectively.

However, a crucial drawback [9] of the standard Relief algorithm is that it lacks mechanisms to tackling outliers and redundant features, which heavily degrade its performance in practice.

- The success of Relief hinges largely on its attempting to discriminate between neighboring patterns (nearest-miss and nearest-hit). However, the nearest neighbors are defined in the original feature space. When there are a large number of redundant and/or noisy features present in the data, it is less likely that the nearest neighbors in the original feature space will be the same as those in the target feature space. As a consequence, the performance of Relief can be degraded drastically;
- The objective function of the Relief algorithm, Eq.(1), is to maximize the average margin of the training samples. This formulation makes it rather vulnerable to outliers, because the margins of outlying patterns usually take very negative values (thus can heavily affect the performance of Relief).

The recently proposed Iterative-Relief algorithm (I-Relief, [9]) addresses these two problems by introducing three latent variables for each pattern and employing the Expectation-Maximization (EM) principal to optimize the objective function. Powerful as it is, this algorithm surfers two drawbacks: (*i*) It is very time-consuming since there is no close-form solution. Therefore, iterative optimization scheme must be employed. In particular, within each iteration, the I-Relief algorithm involves at least $O(N^2D)$ times of computation, which is only tractable for very small data set; (*ii*) I-Relief requires storing and manipulating three $N \times N$-sized matrix at each iteration, which is infeasible for large data set.

In this paper, we propose efficient alternative approaches to address the deficiencies of Relief in tackling outliers and noises. In particular, in order to handle outliers, we borrow the concept of margin-based loss function [1,6] from the supervise learning literature, and integrate a loss function into the objective function of Relief, i.e.: instead of maximizing the average margin, this method minimizes the empirical sum of a specific loss function. Since the resulted problem has a close-form solution, this method is much more efficient (in fact, it is of the same complexity as the standard Relief). In the meanwhile, when appropriate loss functions are chosen, this method can achieve comparable performance as I-Relief. In addition, to tackling noisy features, we propose a novel algorithm, named Exact-Relief, which is based on a new perspective of Relief as a greedy nonparametric Bayes error minimization feature selection approach. We finally conduct empirical evaluations on various benchmark information retrieval tasks. The results confirm the advantages of our proposed algorithms.

2 The Proposed Algorithms

2.1 Against Outliers: Ramp-Relief

Relief maximizes the empirical average margin on the training set (see Eq.(1)). An alternative (and equivalent) way to view this is to minimize the empirical sum of a margin-based loss function:

$$\begin{array}{l} \min \ \sum_{n=1}^{N} l(\mathbf{w}^T \mathbf{m}_n) \\ s.t. : ||\mathbf{w}|| = 1, w_d \geq 0, d = 1, 2, ..., D \end{array} \tag{2}$$

where $l(\cdot)$ is a margin-based loss function [1,6]. In this viewpoint, the standard Relief is a special case of the above formulation, i.e., it uses a simple linear loss function $l(z)$=-z.

To minimize empirical sum of a specific margin-based loss function has been extensively studied in supervised learning literature both theoretically and empirically. This methodology offers various advantages. We refer the interested readers to [6,1] and the references therein for more detailed discussions.

The new perspective of Relief allows us to extend Relief from using linear loss function to other more extensively studied loss functions. For computational simplicity, we solve an approximate problem in this paper, i.e.:

$$\begin{array}{l} \min \ \sum_{n=1}^{N} \mathbf{w}^T l(\mathbf{m}_n) \\ s.t. : ||\mathbf{w}|| = 1, w_d \geq 0, d = 1, 2, ..., D \end{array} \tag{3}$$

and a variation of the Ramp loss function used in ψ-learning [8] is employed:

$$\begin{aligned} r(z) &= \max(z_2, \min(z_0 - z, z_1)) \\ &= \begin{cases} z_1, z < z_0 - z_1 \\ z_2, z > z_0 - z_2 \\ z_0 - z, else \end{cases} \end{aligned} \tag{4}$$

where z_0, z_1 and z_2 are three constants. By using the Lagrangian technique, a quite simple close-form solution to problem Eq.(4) can be easily derived, i.e.:

$$\mathbf{w} = (\boldsymbol{\gamma})^+ / ||(\boldsymbol{\gamma})^+|| \tag{5}$$

where $\boldsymbol{\gamma} = \sum_{n=1}^{N} -r(|\mathbf{x}_n - H(\mathbf{x}_n)| - |\mathbf{x}_n - M(\mathbf{x}_n)|)$, and $(\cdot)^+$ denotes the positive part.

We term this algorithm as **Ramp-Relief** (**R-Relief**). We will show that the R-Relief algorithm is able to deal with outliers as well as I-Relief but is much more efficient and simpler to compute.

2.2 Against Noisy Features: Exact-Relief

Recently, we found that Relief greedily attempts to minimize the nonparametric Bayes error estimated by k-nearest-neighbor (kNN) methods with feature

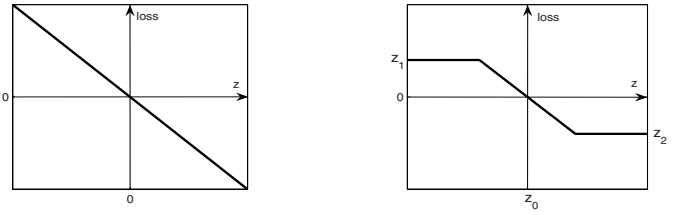

Fig. 1. Linear loss function (left) and ramp loss function (right)

weighting as the search strategy [10]. One of the assumptions made by Relief is that the nearest neighbor of a pattern $\mathbf{x}$ locates close to $\mathbf{x}$ in any single dimensional space. For instance, suppose $\mathbf{x}_a$ is the nearest neighbor of $\mathbf{x}$: $||\mathbf{x}_a - \mathbf{x}|| \leq ||\mathbf{x}_n - \mathbf{x}||$ for $n = 1, 2, \ldots, N$, Relief implicitly assumes that $x_a^{(d)}$ is also, approximately, the nearest neighbor of $x^{(d)}$, that is $x_a^{(d)} \approx x_b^{(d)}$, where $|x_b^{(d)} - x^{(d)}|| \leq ||x_n^{(d)} - x^{(d)}||$ (b is dependent on d). Therefore, Relief approximates $x_b^{(d)}$ with $x_a^{(d)}$ for all $d = 1, 2, \ldots, D$. Although this approximation can reduce the computation complexity significantly, it also pays prices. In particular, if the feature set is strongly redundant such that a large proportion of features are irrelevant, noisy, or useless. In that case, $x_a^{(d)}$ is highly unlikely to locate close to $x^{(d)}$, which can heavily degrade the performance of the solutions. Therefore, it may be preferable to eliminate this assumption. For this purpose, we propose an algorithm refereed as '**Exact-Relief**' (**E-Relief**), which resemble the standard Relief algorithm except using a different margin definition: $\mathbf{m}_n = (m_n^{(d)})_{D\times 1}$, $m_n^{(d)} = |x_n^{(d)} - M_n^{(d)}| - |x_n^{(d)} - H_n^{(d)}|$, where $M_n^{(d)}$ and $H_n^{(d)}$ denote the nearest-miss and nearest-hit of $\mathbf{x}_n$ in the d-th dimension.

2.3 Against Both Outliers and Noisy Features

In practice, it is quite possible that both outliers and noisy features are present in the data. For instance, in spam filtering, junk mails usually contain a large amount of noisy characters in order to cheat the filter. On the other hand, legitimate mails may only have very few words but contain many hyperlinks. Such mails not only contain many noisy features but can also be easily detected as outliers. To handle both factors, an obvious strategy is to combine the R-Relief and E-Relief algorithm, i.e.:

$$\begin{aligned} &\max \sum_{d=1}^{D} w_d \sum_{n=1}^{N} r(|x_n^{(d)} - M_n^{(d)}| - |x_n^{(d)} - H_n^{(d)}|) \\ &s.t. : \mathbf{w} \geq \mathbf{0}, ||\mathbf{w}|| = 1 \end{aligned} \tag{6}$$

We term this algorithm as **ER-Relief**. It can be easily seen that ER-Relief and E-Relief are of the same complexity, i.e., $O(N^2 D)$, which is much more efficient compared to I-Relief, whose worst case complexity is $O(N^3 D)$.

Table 1. Characteristics of data sets

Data Set	#Train	#Test	#Feature	#Class
Spam	1000	3601	57	2
LRS	380	151	93	48
Vowel	530	460	11	11
Trec11	114	300	6429	9
Trec12	113	200	5799	8
Trec23	84	120	5832	6
Trec31	227	700	10127	7
Trec41	178	700	7454	10
Trec45	190	500	8261	10

3 Experiments

In this section, we conduct extensive experiments to evaluate the effectiveness and efficiency of the proposed methods in comparison with state-of-art algorithms in Relief family.

3.1 Experiments on UCI Data Sets

To demonstrate the performance of the proposed algorithms in different information retrieval tasks, we first perform experiments on three benchmark UCI data sets, namely, the spam filtering data set (`Spam`), the low-resolution satellite image recognition data set (`LRS`) and the speaker-independent speech recognition data set (`Vowel`). To conduct comparison in a controlled manner, fifty irrelevant features (known as 'probes') are added to each pattern, each of which is an independently Gaussian distributed random variable, i.e., $\mathcal{N}(0,20)$. The efficiency of a feature selection algorithm can be directly measured by its running time. To evaluate the effectiveness, two distinct metrics are used. One is the classification accuracy estimated by kNN classifier, where k is determined by five-fold cross validation. The other metric is the Receiver Operating Characteristic (ROC) curve [9], which is used to indicate the abilities of different feature selection algorithms in identifying relevant features and at the same time ruling out useless ones. To eliminate statistical deviations, all the experiments are repeated for 20 runs. In each run, the data set is randomly partitioned into training and testing data, and only the training data are used to learn the feature selector. Three groups of experiments have been done:

1. **Against outliers.** Relief, I-Relief and R-Relief are compared. A randomly selected subset of 10% training samples are mislabelled. The testing data is kept intact. No probe is added. The testing errors is shown in the top line of Fig.2. We can see R-Relief improves the performance of Relief significantly. It performs comparably with I-Relief when outliers are present.
2. **Against noisy features.** E-Relief is compared with Relief and I-Relief. 50 probes are added to each example, but no mislabelling is conducted. The

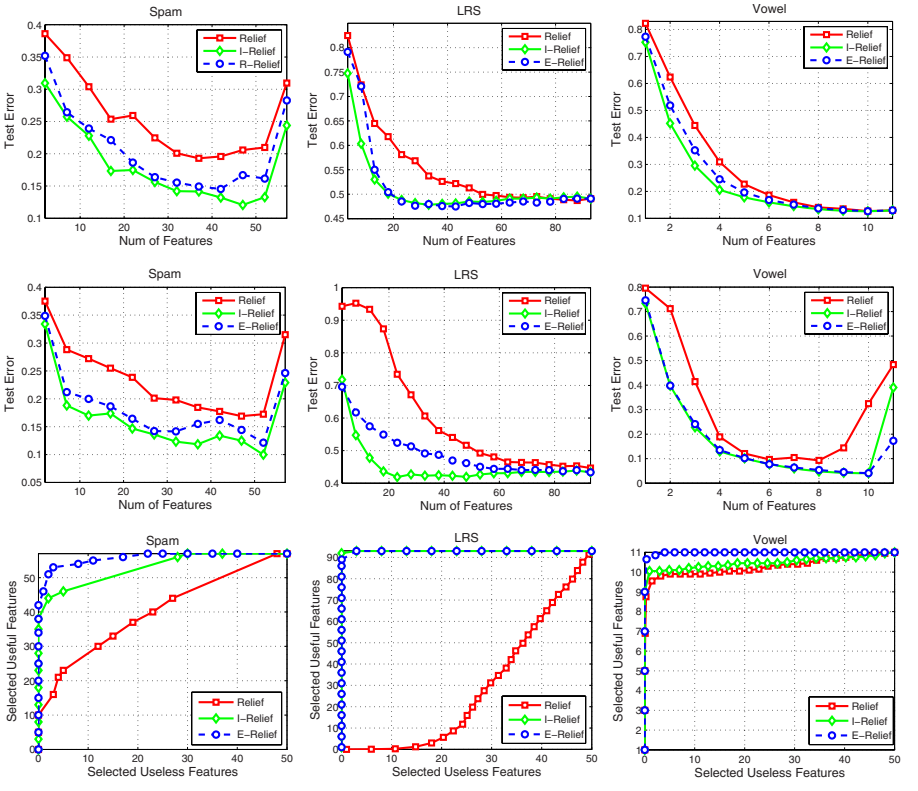

Fig. 2. Comparison of R-Relief/E-Relief I-Relief and Relief on UCI data set

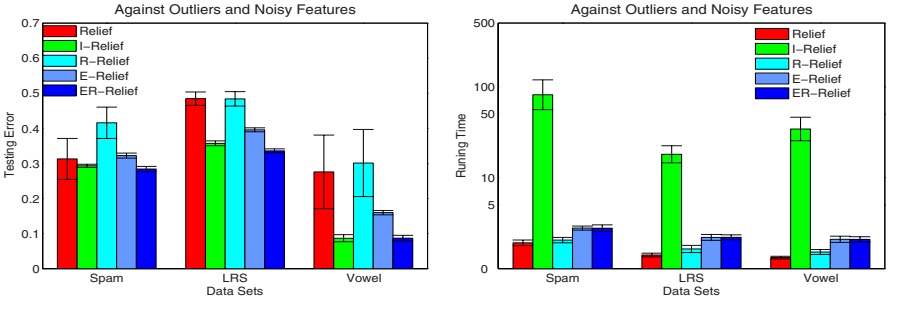

Fig. 3. Average testing errors and running times (sec.) as well as standard deviations on UCI data sets when both outliers and noisy features are involved

testing errors are shown in Fig.2 (middle line). The ROC curves are also plotted in Fig.2(bottom line). We can see that E-Relief performs comparably with I-Relief (much better than Relief) when noisy features are involved.

3. **Against outliers and noisy features.** E-Relief, R-Relief and their combination, ER-Relief, along with I-Relief and Relief are compared. 50 probes are added to each pattern (both training and testing), and 10% of training samples are mislabeled. The testing errors and running times of each algorithm, with average values and standard deviations, are shown by two bar plot, in Fig.3. We can see that in the presence of both outliers and noisy features, the performance of Relief is degraded badly. R-Relief and E-Relief do not necessarily improve the performance. However, their combination, ER-Relief, improves the performance drastically. In most cases, ER-Relief performs comparably with I-Relief. In some cases, it performs the best. With respect to computational efficiency, we can see that E-Relief, R-Relief and ER-Relief do not introduce a large increase of computational expense compared to Relief, while I-Relief is far more time-consuming.

3.2 Document Clustering and Categorization

We then apply the algorithms to document clustering and categorization tasks. For this purpose, six benchmark text data sets from **Trec** (the Text REtrieval Contest, http://trec.nist.gov) collection that are frequently used in information retrieval research are selected. The information of each data set is also summarized in Table.1.

The Relief, I-Relief and ER-Relief algorithms are compared, with no probe or mislabelling. For text clustering, C-mean algorithm is employed to get the clustering result after dimensionality reduction. For simplicity, the number of cluster, C, is set to be the true number of classes. For document categorization, the nearest-neighbor classifier is applied for final classification. Each experiment is repeated for 20 runs, each of which is based on a random splitting of the data set. The $Macro_{ave}F_1$ and $Micro_{ave}F_1$ are used to assess the classification results, and ARI (Adjusted Rand Index) and NMI (Normalized Mutual Information) are used to evaluate the clustering results. Table.2 presents the best average result of each algorithm.

Again, we observe that (i) ER-Relief performs much better than Relief, and that (ii) ER-Relief has achieved comparable performances comparably to I-Relief in most cases, although its computation complexity and operating time are much less than I-Relief. Note that the results about the running time are not given due to space limitation.

In information retrieval, *huge amount of data* and *extremely high dimensionality* are two core challenges (and are also becoming increasingly challenging). Therefore, the efficiency of ER-Relief as well as its effective ability to identify a small subset of predictive features (out of a huge amount of redundant ones) may make it a rather appealing and encouraging tool for both challenges, i.e., it is efficient with respect to data set size, and, it is able to effectively reduce the dimensionality. This confirms our attempting in applying ER-Relief to information retrieval tasks and encourages us to investigate its performance in extensive IR applications in the future.

Table 2. Comparison of feature weighting algorithms: Relief (RLF), I-Relief (IRLF) and ER-Relief (ERRL), in text categorization and clustering tasks. Best results are highlighted in bold.

	Categorization						**Clustering**					
	$Macro_{avg}F_1$			$Micro_{avg}F_1$			*ARI*			*NMI*		
	RLF	IRLF	ERRL	RLF	IRLF	ERRL	RLF	IRLF	ERRL	RLF	IRLF	ERRL
Trec11	**0.50**	0.43	0.45	**0.61**	0.54	0.57	**0.17**	0.13	0.11	**0.25**	0.16	0.16
Trec12	0.58	0.58	**0.64**	0.59	0.58	**0.60**	0.04	0.06	**0.07**	0.10	0.13	**0.15**
Trec23	0.49	**0.53**	0.42	0.62	**0.66**	0.59	0.04	**0.07**	0.05	0.09	**0.12**	0.10
Trec31	0.66	0.66	**0.71**	0.82	0.80	**0.86**	0.07	0.08	**0.09**	0.13	0.12	**0.15**
Trec41	0.65	**0.68**	0.64	0.74	0.77	**0.78**	0.16	**0.17**	**0.17**	0.20	**0.33**	0.29
Trec45	**0.63**	0.54	0.61	0.68	0.61	**0.71**	0.06	0.05	**0.07**	0.19	0.16	**0.21**

4 Conclusion

Fast growing internet data poses a big challenge for information retrieval. Feature selection, for the purpose of defying curse of dimensionality among others, plays a fundamental role in practice. Relief is an appealing feature selection algorithm. However, it lacks mechanisms to handle outliers and noisy features. In this paper, we have established two algorithms to address these two factor respectively. Compared with the recently proposed I-Relief, our algorithms are able to achieve comparable performance, while operating much more efficiently, which is proved by extensive experiments on various benchmark information retrieval tasks.

References

1. Bartlett, P., Jordan, M.I., McAuliffe, J.D.: Convexity, Classification and Risk Bounds. J. of American Stat. Assoc. 101(473), 138–156 (2006)
2. Dash, M., Liu, H.: Feature Selection for Classification. Intelligent Data Analysis (IDA) 1, 1131–1156 (1997)
3. Guyon, I., Elissee, A.: An Introduction to Variable and Feature Selection. JMLR 3, 1157–1182 (2003)
4. Hall, M.A., Holmes, G.: Benchmarking Attribute Selection Techniques for Discrete Class Data Mining. IEEE Trans. KDE 15(3), 1437–1447 (2003)
5. Kira, K., Rendell, L.A.: A Practical Approach to Feature Selection. In: Proc. of Ninth ICML, pp. 249–256 (1992)
6. Lin, Y.: A Note on Margin-based Loss Functions in Classification. Statistics and Probability Letters 68, 73–82 (2004)
7. Robnik-Šikonja, M., Kononenko, I.: Theoretical and Empirical Analysis of ReliefF and RRlief. J. Machine Learning 53(1-2), 23–69 (2003)
8. Shen, X., Tseng, G., Zhang, X., Wang, W.: On ψ-learning. J. of American Stat. Assoc., 724–734 (1998)
9. Sun, Y.J.: Iterative Relief for Feature Weighting: Algorithms, Theories, and Applications. IEEE Trans. PAMI 29(6), 1035–1051 (2007)
10. Yang, S.H., Hu, B.G.: Feature Selection by Nonparametric Bayes Error Minimization In: Proc. of the 12th PAKDD (2008)