CURRENT AND FUTURE TRENDS IN FEATURE SELECTION AND EXTRACTION FOR CLASSIFICATION PROBLEMS

LAWRENCE B. HOLDER
Department of Computer Science and Engineering, University of Texas at Arlington, TX, USA
holder@cse.uta.edu

INGRID RUSELL
Department of Computer Science, University of Hartford, CT, USA
irussell@hartford.edu

ZDRAVKO MARKOV
Department of Computer Science, Central Connecticut State University, CT, USA
markov@ccsu.edu

ANTHONY G. PIPE
School of Electrical and Computer Engineering, University of West of England, UK
Anthony.Pipe@uwe.ac.uk

BRIAN CARSE
School of Electrical and Computer Engineering, University of West of England, UK
Brian.Carse@uwe.ac.uk

In this article, we describe some of the important currently used methods for solving classification problems, focusing on feature selection and extraction as parts of the overall classification task. We then go on to discuss likely future directions for research in this area, in the context of the other articles from this special issue. We propose that the next major step is the elaboration of a theory of how the methods of selection and extraction interact during the classification process for particular problem domains, along with any learning that may be part of the algorithms. Preferably this theory should be tested on a set of well-established benchmark challenge problems. Using this theory, we will be better able to identify the specific combinations that will achieve best classification performance for new tasks.

Keywords: Classification, feature selection, feature extraction

1. Introduction

A common theme among the articles in this special issue is the use of feature selection and feature extraction to improve the performance on a classification task, more specifically, improving a learning algorithm's ability to identify a hypothesis that more closely approximates the target function. In order to place these methods in the context of the field, we here discuss the prevalent approaches to feature selection and extraction for classification. In general, the classification task involves a set of m examples $\{(\vec{x},y)\}$, where $\vec{x}$ is a vector of n features $<x_1, x_2, ..., x_n>$ and y is a class, and a target function $f(\vec{x})=y$ that maps an example's feature vector to its class. The goal of the classification task is to identify a hypothesis h that approximates f so as to minimize the classification error, e.g., $\sum_{i=1}^{m}(f(\vec{x}_i)-h(\vec{x}_i))^2$. The goal of feature *selection* is to identify a new feature vector $\vec{w}$ that is a subset of the set of features

$\{x_1, x_2, \ldots, x_n\}$ and that when used by the learning algorithm, yields a hypothesis h with less error and/or in less time. The goal of feature *extraction* is to transform $\vec{x}$ into a new feature vector $\vec{z}$ that, when used by the learning algorithm, yields a hypothesis h with less error and/or in less time. There are many variations on this framework, but this one will provide sufficient context in which to discuss different approaches to feature selection, feature extraction and classification. We will take each of these topics in turn followed by a discussion of the specific applications represented by the articles in this special issue.

2. Feature Selection

Numerous methods exist for identifying the subset of features $\vec{w}$ that will ultimately improve performance on the classification task. In this section we discuss some of these methods, especially those related to the methods described in the articles in this special issue. The first group of feature selection methods involves filtering the original features or ranking them with the expectation that the learning algorithm will take advantage of the rank to reduce the dimensionality of the learning task. Both these methods utilize a metric for determining the relevance or importance of each feature. Two popular metrics are correlation and mutual information. The correlation of a feature x_i to the class y is expressed as the ratio of their covariance (*cov*) divided by the square root of the product of their variances (*var*):

$$\frac{\mathrm{cov}(x_i, y)}{\sqrt{\mathrm{var}(x_i)\,\mathrm{var}(y)}}$$

The mutual information between a feature x_i and the class y can be expressed as the sum over all possible values of x_i and y of the following:

$$p(x_i, y) \log \frac{p(x_i, y)}{p(x_i)\,p(y)}$$

Either the correlation or mutual information measure can be used to filter or rank the set of features.

While filtering and ranking methods have demonstrated performance improvements for feature selection, Zhang's article (Ref. 17) in this issue provides some theoretical results that question the general usefulness of these methods when using a naïve Bayes classifier in the presence of conditional dependence between features.

A second group of feature selection methods uses the learning algorithm to help identify the subset of features. Embedded methods[8] consider different subsets of the features during the learning process, where the subsets are evaluated based on their ability to support correct classification of the training examples. Bisant's article[1] in this issue uses embedded feature selection by modifying the objective function of a

neural network so that in addition to learning the weights of the links in the network, his method also learns weightings for the features. A similar approach is taken in Ref. 12 using support vector machines as the underlying learning algorithm. The alternative to the embedded approach is the wrapper approach[9] in which the learning algorithm is "wrapped" by a search strategy that performs a search through the space of feature subsets, where each subset $\vec{w}$ is evaluated based on the performance of the hypothesis learned using by the learning algorithm using only the features in $\vec{w}$.

In both the embedded and wrapper approaches the space of possible subsets can be considered from different directions. In *forward selection*, the approach begins with a small subset and adds additional features to the subset if they improve the performance of the learned hypothesis. Alternatively, *backward elimination* begins with nearly all the original features and eliminates features as long as there is no reduction in the performance of the learning hypothesis. Both directions have advantages and disadvantages, and the correct choice typically depends on the domain of the classification task.

3. Feature Extraction

In many classification task domains the given features are not sufficient to achieve acceptable classification performance, but a transformation of the features may yield new features that are more highly correlated with the class value. In this section we describe several methods for extracting features from the original feature set. The first set of methods we include in this area of feature extraction is various pre-processing steps that can be performed on the examples that modify the original feature values. Such methods include discretization, in which a continuous-valued feature is mapped to a discrete set of ranges, and normalization, in which feature values are transformed into a pre-specified range of values. Also within the realm of a pre-processing approach are the methods that identify a new dimension of the feature space which is defined by a linear or non-linear combination of the original features. Examples of these pre-processing methods include Principal Component Analysis (PCA), Singular Value Decomposition (SVD), and various forms of clustering (see Ref. 5). Yu et al.'s article (Ref. 15) in this issue describes a non-singular approach based on the Karhunen-Loeve Transform (KLT).

Some of the articles in this special issue use approaches to feature extraction more specialized to the domain, the hypothesis class, or the learning algorithm. In the domain of text classification, Zelikovitz and Marquez[16] use Latent Semantic Indexing (LSI, see Ref. 2) as an initial step to transforming the text features into semantic correlations before performing a SVD to identify new extracted features. Zhong[18] uses Hidden Markov Models (HMMs) as the class of hypotheses to learn, effectively transforming the original features into a HMM "feature," which is then used for sequence classification. Lastly, while Yu et al.[15] do show success with their KLT method for improving the performance of a multi-layer perceptron (MLP) neural network, they also show that other common linear transform approaches, like SVD, may not have a beneficial effect on MLP performance, since they may merely achieve the equivalent of a different set of initial network weights.

One more recent feature extraction method that deserves mention here applies in relational domains. From the outset we have assumed that the training data is in the form of a feature vector. However, in relational domains, there can be relational features that describe a relationship between examples. For instance, we can use feature vectors to describe attributes of people, but to express arbitrary relationships between the people, a feature vector would not suffice. Unfortunately, most of the methods for learning in relational domains are considerably more computationally complex than learning with non-relational (i.e., propositional) feature vector data. For this reason a feature extraction method called *propositionalization* (see Ref. 10) has been developed in which salient relational features are transformed into components of a propositional feature vector so that simpler learning algorithms can be applied. However, the challenge is to maintain classification performance despite the loss of information.

4. Classification

The success of the aforementioned feature selection and extraction methods depends on the ability of the learning algorithm to effectively utilize the modified set of features. Several of the articles in this special issue describe the positive effect these methods have on classification performance for a number of learning algorithms. In this section we describe the learning methods used within these papers as well as some other popular learning methods used in the context of feature selection and extraction. First, Zhang[17] provides a theoretical understanding of when naïve Bayes will perform optimally in terms of the conditional independence of the features, and he describes a new algorithm that helps overcome those situations leading to sub-optimal performance of naïve Bayes. Zhang's article presents a theoretical contribution towards better understanding of this very popular and accurate classification approach, which is a type of Bayesian (belief) Network. The Naïve Bayes algorithm has been intensively studied, both theoretically and experimentally. However, the success of the algorithm is mostly based on its good performance, rather than good theoretical understanding of its operation.

Neural networks are among the most popular learning algorithms because of their generality and classification speed. Bisant[1] describes an application of a neural network combined with embedded feature selection to perform document genre identification. Ferguson et al.[6] perform pre-processing of the data using clustering to identify a modular neural network, that is, a set of networks each trained on a specific subset of the data. The modular networks are more accurate and easier to train, and their outputs can be used by another decision module, such that the modular approach outperforms the approach of using one large network. Although Yu et al.[15] show a lack of benefit for pre-processing feature extraction with multi-layer perceptron (MLP) neural networks, they also provide an alternative training algorithm that improves upon this limitation. Where many researchers are attempting to improve training success by dealing with hidden and output weights separately, Hessian matrix-related approaches seem to hold promise compared with other

Newton-related methods. Yu et al's article in this special issue discusses this with respect to non-singular pre-processing techniques (see Ref.15).

Decision tree induction methods (see Ref. 11) are also among the most popular learning algorithms. While none of the articles in this issue specifically analyze the use of these methods in the context of feature selection and extraction, decision-tree induction methods are a common technique used in the wrapper approaches mentioned earlier and ensembles of learning algorithms mentioned below. Ferguson et al.'s article (Ref. 6) in this issue shows how the use of decision tree induction can provide a more effective decision module for their modular neural network techniques. Bisant[1] also uses a decision tree induction algorithm as a basis of comparison to show the superiority of his neural network based approach to sequence analysis and genre identification.

Two other articles in this issue describe methods for taking advantage of unlabeled test cases to help improve feature extraction methods. Semi-supervised learning, a relatively new methodology, is explored in the articles by Zelikovitz & Marquez and by Zhong. It is an alternative to the classical supervised setting for machine learning and allows unlabeled data to be used in the training process. Semi-supervised learning is becoming popular because it helps increase classification accuracy without the need for additional labeled data. In this respect it is suitable for areas where labeling is difficult or requires substantial human effort.

Semi-supervised learning is studied intensively within numeric and statistical approaches to learning. Combined use of labeled and unlabeled data is discussed in the area of clustering where class labels may be used along with the cluster labels to evaluate and improve clustering. Early approaches to semi-supervised learning were associated with the Expectation-maximization (EM) algorithm (see Ref. 4), where mixture models can be created using both labeled and unlabeled data. Semi-supervised learning became especially popular within the statistical learning community after the recent work of Vapnik[14] on transductive inference as an alternative to induction and deduction. Zelikovitz and Marquez[16] utilize transductive learning (see Ref. 14) to improve their LSI-SVD approach for text classification. Their results show that this approach is typically better than simply adding more training data. Zhong[18] uses unlabeled test cases along with the training examples to help train Hidden Markov Models (HMMs) for sequence classification. Again, substantial improvements in performance were shown when the test cases are included in the learning phase.

Finally, we mention two other classification learning algorithms that have shown promising performance over earlier methods and make heavy use of feature selection and extraction methods. First, kernel methods and Support-Vector Machines (SVMs, see Ref. 13) rely on the ability to map the original set of features into a higher-dimensional non-linear space with the hope that simple learning methods in the higher dimension space can quickly find a hypothesis capable of distinguishing the classes. The benefit of kernel methods is that they do not have to actually perform the mapping to the higher-dimensional space, but can perform the same computations using only the lower-dimensional features. Second, recent results have shown that learning several hypotheses using different learning algorithms and

combining their individual classifications can greatly improve performance over relying on a single learning algorithm and a single learned hypothesis. Such approaches are called *ensemble* methods (see Ref. 3). While the ensemble can consist of hypotheses learned using several different learning approaches (e.g., neural network, decision tree induction, etc.) another approach called *bagging* uses the same learning algorithm, but different hypotheses learned using different samples (or bags) of the training examples. One perspective on the ensemble is that it is a collection of selected or extracted features as determined by the different learning algorithms on different samples of the training data.

5. Applications

Finally, we discuss the various application domains represented by the articles in this issue. There are challenging Machine Learning applications, such as gene analysis, EEG analysis and other bioinformatics applications where obtaining a class label is very difficult. The rapidly growing area of Data Mining uses various combinations of supervised and unsupervised approaches where the semi-supervised learning model plays an important role. In text and web mining only a small portion of the large volumes of text or hypertext documents that is being processed is labeled, which makes the use of combined supervised and unsupervised approaches appealing.

Zelikovitz and Marquez[16] apply their transductive-LSI-SVD learning approach to the task of short-text classification, e.g., determining the category of a technical paper based only on the title. This is an increasingly popular Machine Learning application. Its recent popularity is due to the important role it plays in the aforementioned Data and Web Mining, where text mining is a central issue. The classical text classification is based on the vector-space model studied in the area of information retrieval. The basic challenge for this model is the large number of terms (features) compared to a relatively small number of documents (examples). This is a difficult supervised learning task. The article discusses another challenging problem - classifying short text documents which, given the large term space, are difficult to classify too, as they may not share any common terms but still be semantically similar. One of the areas of research that addresses the latter problem is based on using various techniques to transform the original terms space, so that the documents are better separated. Zelikovitz & Marquez propose an approach based on Latent Semantic Indexing (LSI). As LSI was originally discussed within the area of unsupervised learning (clustering) the authors are extending it with an approach to using labeled data as well.

In a similar application Bisant[1] applies neural networks embedded with feature selection to the task of categorizing documents in to various classes of genres. Here, the documents are expressed as a long sequence of text, which motivated the author to apply this approach borrowed from the similar task in molecular sequence analysis. Ferguson et al.[6] apply their clustering pre-processing, modular neural network approach to the task of character recognition. Their self-organizing map approach to clustering identified clear subgroups of characters that were used to train

different neural network modules, and they were able to achieve better performance than an approach using one large network to recognize all twenty-six characters.

Lastly, Zhong[18] applies the semi-supervised HMM-based sequence classification method to classifying EEG series into one of two classes of patients: normal or alcoholic.

While the above applications clearly demonstrate the benefits of feature selection and extraction methods for classification, it is difficult to compare the different approaches. To this end we would like to point out the existence of a set of benchmark challenge problems constructed as part of a feature selection/extraction challenge for the 2003 Workshop on Feature Extraction and Feature Selection Challenge of the Neural Information Processing Symposium (NIPS, see Ref. 7). To support better understanding of the strengths and weaknesses of the various approaches, we recommend that future efforts report performance on these datasets.

6. Conclusions

We have discussed a number of methods in the context of feature selection and extraction for classification, most of which were motivated from the approaches taken in the articles from this special issue. While these articles each evaluate a particular combination of these three aspects, there are many more combinations to investigate. The next major step in these areas is the elaboration of a theory of how these methods interact for particular domains so that, given a particular classification task, we can better identify the specific combinations of feature selection, feature extraction and learning methods that will achieve maximum classification performance. Further investigation of the, as yet unexplored combinations, as well as larger systematic explorations of the method space using standardized challenge problems, will help us to achieve these aims.

References

1. D. Bisant, "An Application of Neural Networks to Sequence Analysis and Genre Identification," *International Journal of Pattern Recognition and Artificial Intelligence* (this issue), 2005.
2. S. Deerwester, S. Dumais, G. Furnas and T. Landauer, "Indexing by Latent Semantic Analysis," Journal for the American Society of Information Science, 41(6):391-407, 1990.
3. T. Dietterich, "Ensemble Methods in Machine Learning," in J. Kittler and F. Roli (Ed.) *First International Workshop on Multiple Classifier Systems, Lecture Notes in Computer Science*, Springer Verlag, 2000.
4. A. Dempster, N. Laird and D. Rubin, Maximum likelihood from incompletedata via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38, 1977.
5. R. Duda, P. Hart and D. Stork, *Pattern Classification, Second Edition*, Wiley, 2000.
6. B. Ferguson, R. Ghosh and J. Yearwood, "Modular Neural Network Design for the Problem of Alphabetic Character Recognition," *International Journal of Pattern Recognition and Artificial Intelligence* (this issue), 2005.

7. I. Guyon (Chair), *Workshop on Feature Extraction and Feature Selection Challenge*, Neural Information Processing Symposium (NIPS), http://clopinet.com/isabelle/Projects/NIPS2003/, 2003.
8. I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection," *Journal of Machine Learning Research*, Issue 3, 2003.
9. R. Kohavi and G. John, "Wrappers for Feature Selection," *Artificial Intelligence*, 97(1-2):273-324, December 1997.
10. S. Kramer, N. Lavrac and P. Flach, "Propositionalization Approaches to Relational Data Mining," in *Relational Data Mining*, S. Dzeroski (Ed.), Springer-Verlag, 2001.
11. J. Quinlan, "Induction of Decision Trees," *Machine Learning*, 1(1):81-106, 1986.
12. A. Rakotomamonjy, "Variable Selection Using SVM-based Criteria," *Journal of Machine Learning Research*, Issue 3, 2003.
13. B. Schoelkopf and A. Smola, *Learning with Kernels – Support Vector Machines, Regularization, Optimization and Beyond*, MIT Press, 2002.
14. V. Vapnik, *Statistical Learning Theory*, John Wiley and Sons, New York, 1998.
15. C. Yu, M. Manry and J. Li, "Effects of Nonsingular Pre-Processing on Feed Forward Network Training," *International Journal of Pattern Recognition and Artificial Intelligence* (this issue), 2005.
16. S. Zelikovitz and F. Marquez, "Transductive Learning for Short Text Classification Problems Using Latent Semantic Indexing," *International Journal of Pattern Recognition and Artificial Intelligence* (this issue), 2005.
17. H. Zhang, "Toward Understanding the Optimality of Naïve Bayes," *International Journal of Pattern Recognition and Artificial Intelligence* (this issue), 2005.
18. S. Zhong, "Semi-Supervised Sequence Classification with HMMs," *International Journal of Pattern Recognition and Artificial Intelligence* (this issue), 2005.