Pattern Synthesis in SVM Based Classifier

C. Radha
Indian Institute of Science, India

M. Narasimha Murty
Indian Institute of Science, India

INTRODUCTION

An important problem in pattern recognition is that of pattern classification. The objective of classification is to determine a discriminant function which is consistent with the given training examples and performs reasonably well on an unlabeled test set of examples. The degree of performance of the classifier on the test examples, known as its generalization performance, is an important issue in the design of the classifier. It has been established that a good generalization performance can be achieved by providing the learner with a sufficiently large number of discriminative training examples. However, in many domains, it is infeasible or expensive to obtain a sufficiently large training set. Various mechanisms have been proposed in literature to combat this problem. Active Learning techniques (Angluin, 1998; Seung, Opper, & Sompolinsky, 1992) reduce the number of training examples required by carefully choosing discriminative training examples. Bootstrapping (Efron, 1979; Hamamoto, Uchimura & Tomita, 1997) and other pattern synthesis techniques generate a synthetic training set from the given training set. We present some of these techniques and propose some general mechanisms for pattern synthesis.

BACKGROUND

Generalization performance is generally quantified using the technique of structural risk minimization (Vapnik, 1998). The risk of misclassification is dependent on the training error, the V-C dimension of the classifier and the number of training examples available to the classifier. A good classifier is one which has a low training error and low V-C dimension. The generalization performance of the classifier can also be improved by providing a large number of training examples.

The number of training examples required for efficient learning depends on the number of features in each training example. Larger the number of features, larger is the number of training examples required. This is known as the curse of dimensionality. It is generally accepted that at least ten times as many training examples per class as the number of features are required (Jain & Chandrasekharan, 1982).

However, in most applications, it is not possible to obtain such a large set of training examples. Examples of such domains are:

1. **Automated Medical diagnosis:** This involves designing a classifier which examines medical reports and determines the presence or absence of a particular disorder. For efficient classification, the classifier must be trained with a large number of positive and negative medical reports. However, an adequate number of medical reports may not be available for training because of the possible high cost of performing the medical tests.
2. **Spam Filtering:** A good mail system aims at identifying and filtering out spam mails with minimum help from the user. The user labels mails as spam as and when she encounters them. These labeled mails are used to train a classifier which can further be used to identify spam mails. For the mail system to be useful, the user should be presented with the least possible number of spam mails, which calls for an efficient classifier. However, sufficient training examples are not available to the learner to perform efficient classification.
3. **Web page recommendations:** The user is provided recommendations based on her browsing history. If good recommendations are to be provided early, then the recommendation system will have to manage with a relatively low number of

examples representing web pages of interest to the user.

There are two major techniques that can be employed to combat the lack of a sufficiently large training set: Active Learning and Pattern Synthesis.

ACTIVE LEARNING

Active Learning is a technique in which the learner exercises some control over the training set. In pool-based active learning, a learner is provided with a small set of labeled training examples and a large set of unlabeled examples. The learner iteratively chooses a few examples from the unlabeled examples for labeling; the user provides the labels for these examples and the learner adds these newly labeled examples to its training set.

In the "membership query" paradigm of active learning, the learner queries the user for the label of a point in the input region (Angluin, 1998).

Active Learning aims at finding a good discriminant function with minimum number of labeled training examples. The version space for a given set of training examples is defined as the set of all discriminant functions that consistently classify the training examples. Version space is the region of uncertainty, the region which contains those examples whose labels the learner is uncertain about. Active Learning methods choose from this region, those unlabeled examples for labeling, which reduce the version space as fast as possible, ideally by half in each iteration.

Various algorithms for active learning have been proposed in literature. Some of them are "Query by Committee" (Seung, Opper, & Sompolinsky, 1992), "Committee Based Sampling" (Dagan, & Engelson, 1995), etc. These techniques train a committee of classifiers and after each iteration of training, the labels of the unlabeled examples are determined individually by each member of the committee. That unlabeled example for which there is maximum disagreement among the committee members is chosen for labeling. Active Learning has been incorporated along with the Expectation – Maximization algorithm to improve efficiency (Nigam, McCallum, Thrun, & Mitchell, 2000). This technique has also been employed in SVM training (Tong & Koller, 2001; Tong & Chang, 2001).

PATTERN SYNTHESIS

In some domains it may not be possible to obtain even unlabeled examples. In some others, the user may not be available or may not be in a position to provide the labels for the unlabeled examples. In especially such circumstances, it is useful to generate artificial training patterns from the given training set. This process is known as pattern synthesis.

Various techniques have been devised in literature to obtain an artificial training set. Some of the techniques are described below in brief:

1. **Bootstrapping:** This technique involves approximating the sample probability distribution from the given examples and drawing random samples from this distribution. This sample is called the bootstrap sample and the distribution is called the bootstrap distribution. The bootstrap distribution can be estimated through various means, including direct calculation, Monte Carlo estimation and Taylor's series expansion (Efron, 1979). This technique has been successfully applied in Nearest Neighbor Classifier design (Hamamoto, Uchimura & Tomita, 1997).
2. **Using Domain-related information:** These techniques involve exploiting domain knowledge such as radial symmetry (Girosi & Chan, 1995) and transformation invariance (Niyogi, Girosi & Poggio, 1998). Such invariances have been applied to the support vectors obtained from a Support Vector Machine (Scholkopf, Burges & Vapnik, 1996). In this technique, an initial model is learnt to determine the support vectors in the training set. The transformations are then applied to these support vectors to obtain synthetic patterns.

MAIN FOCUS

General Pattern Synthesis Techniques

These techniques involve applying a transformation to the given training examples to obtain new training