Evolutionary Feature Space Transformation Using Type-Restricted Generators

Oliver Ritthoff, Ralf Klinkenberg
{*ritthoff,klinkenberg*}*@ls8.cs.uni-dortmund.de*

Chair of Artificial Intelligence, Department of Computer Science,
University of Dortmund, 44221 Dortmund, Germany
`http://www-ai.cs.uni-dortmund.de/`

Abstract. Data preprocessing, especially in terms of feature selection and generation, is an important issue in data mining and knowledge discovery tasks. Genetic algorithms proved to work well on feature selection problems where the search space produced by the initial feature set already contains the target hypothesis. In cases where this precondition is not fulfilled, one needs to construct new features to adequately extend the search space. As a solution to this representation problem, we introduce a framework combining feature selection and type-restricted feature generation in a wrapper-based approach using a modified canonical genetic algorithm for the feature space transformation and an inductive learner for the evaluation of the constructed feature set.

A crucial aspect for successfully solving a learning task at hand is the language in which the hypotheses, i.e. possible solutions, are represented. Two learning tasks that handle the representation problem by properly transforming an inadequate feature space are *feature selection* and *feature generation* [2]. Models of *feature selection* assume that the description language contains a superset of the features that are sufficient to describe the target hypothesis. Thus, learning comprises the selection of a feature subset that maximizes the learning performance in a classification or regression task. There are a number of heuristic feature selection strategies, incrementally choosing feature subsets that lead to the highest performance increase in one iteration. Yet, in contrast to genetic algorithms (GAs) [3], a major shortcoming of such methods is their lacking ability to cope with complex, multimodal search spaces. The main goal of *feature generation* is to reveal feature dependencies that need to be recognized to find the target concept. This is done by transforming the original feature set into a feature set more suitable for the learning task at hand.

The overall approach is structured as follows. The modified GA produces individuals by varying and recombining given feature sets and conducts the search for a good feature set using the learning algorithm for its evaluation. The training data set the learning algorithm is run on is partitioned into internal training and hold out sets, with the feature sets removed from and added to the data that were acquired in the GA search step. The process of creating feature sets, using the modified GA and evaluating these sets is repeated until a given

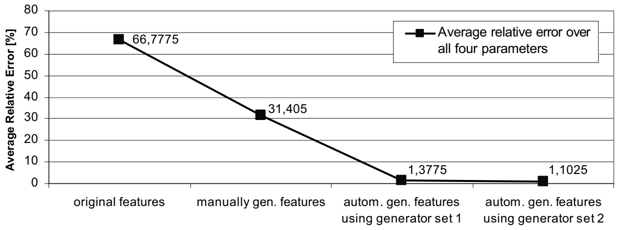

Fig. 1. Average relative error over all four parameters on a two-substance mixture

termination criterion is fulfilled. The resulting feature set is chosen as the final set on which to run the learning algorithm. The final evaluation of the resulting classifier is done using an independent test set not used during the learning step. The feature generation process is based on two classes of feature types, namely the *value type* and the *block type*, together with their ontologies, each describing a hierarchical is-superset-of relation. The *value type* specifies the data type (e.g. nominal or real) of an individual feature, whereas the *block type* contains some meta-data about the feature, e.g. if it is just an individual feature or a part of a time series. The idea is to restrict the constructible features to those matching the required types of the feature generator at hand, e.g. the "plus" generator should only be applied to numeric features, discarding e.g. all nominal features.

The experiment shown in Figure 1, which has been conducted with our flexible learning environment YALE [1], investigates our approach focusing on the aspect of type-restrictions for feature generation in the field of chromatography. The learning task was to predict four characteristic coefficients of a two component mixture given the corresponding chromatogram time series, representing the original feature set. We compared the performance of the presented approach, automatically generating an optimized feature set using different generator sets, with a manually created feature set and the original feature set as baselines for the prediction performance. The first generator set contained the arithmetic generators *plus, minus, multiply*, and *divide* (generator set 1), the second set additionally comprised the generator *time series*, producing several function characteristics (generator set 2). Not adapting the feature space, and even manual feature generation proved to be far less effective than our automatic transformation approach in terms of predictive error. Furthermore, including domain knowledge using explicit ontology-based type-restrictions significantly limits the feature space and thus accelerates the search process.

References

1. S. Fischer, R. Klinkenberg, I. Mierswa, and O. Ritthoff. Tutorial for YALE: Yet Another Learning Environment. Technical Report CI 136/02, SFB 531, University of Dortmund, June 2002. `http://yale.uni-dortmund.de/`.
2. H. Liu and H. Motoda. *Feature Extraction, Construction, and Selection: A Data Mining Perspective.* Kluwer, Dordrecht, NL, 1998.
3. M. Mitchell. *An Introduction to Genetic Algorithms.* MIT Press, 1996.