Search Strategies for Ensemble Feature Selection in Medical Diagnostics

Alexey Tsymbal[*], Pádraig Cunningham[*], Mykola Pechenizkiy[**], Seppo Puuronen[**]

[*]*Department of Computer Science, Trinity College Dublin, Ireland,*
e-mails Alexey.Tsymbal@cs.tcd.ie, Padraig.Cunningham@cs.tcd.ie
[**]*Department of Computer Science and Information Systems, University of Jyväskylä,*
Jyväskylä, Finland, e-mails mpechen@cs.jyu.fi, sepi@cs.jyu.fi

Abstract

The goal of this paper is to propose, evaluate, and compare four search strategies for ensemble feature selection, and to consider their application to medical diagnostics, with a focus on the problem of the classification of acute abdominal pain. Ensembles of learnt models constitute one of the main current directions in machine learning and data mining. Ensembles allow us to get higher accuracy, sensitivity, and specificity, which are often not achievable with single models. One technique, which proved to be effective for ensemble construction, is feature selection. Lately, several strategies for ensemble feature selection were proposed, including random subspacing, hill-climbing-based search, and genetic search. In this paper, we propose two new sequential-search-based strategies for ensemble feature selection, and evaluate them, constructing ensembles of simple Bayesian classifiers for the problem of acute abdominal pain classification. We compare the search strategies with regard to achieved accuracy, sensitivity, specificity, and the average number of features they select.

1. Introduction

Current electronic data repositories, especially in medical domains, contain enormous amounts of data. These data includes also currently unknown and potentially interesting patterns and relations, which can be uncovered using knowledge discovery and data mining methods [5]. These methods were successfully applied in a number of medical domains, e.g. in the localization of a primary tumor, prognostics of recurrence of breast cancer, diagnosis of thyroid diseases, and rheumatology [5].

A popular method for creating an accurate classifier from a set of training data is to train several different classifiers, and then to combine their predictions [4]. An ensemble is often more accurate than any of the single classifiers in the ensemble. Both theoretical and empirical research have demonstrated that a good ensemble is one where the base classifiers in the ensemble are both accurate and tend to err in different parts of the input space (e.g., have high diversity in their predictions). One efficient way to construct an ensemble of diverse classifiers is to use different feature subsets. An important issue in creating an effective ensemble is the choice of the function for combining the predictions of the base classifiers. It was shown that increasing coverage of an ensemble through diversity is not enough to ensure increased prediction accuracy – if the integration method does not utilize coverage, then no benefit arises from integrating multiple models [4].

One effective approach for generating an ensemble of accurate and diverse base classifiers is to use *ensemble feature selection* [9]. By varying the feature subsets used to generate the base classifiers, it is possible to promote diversity and produce base classifiers that tend to err in different subareas of the instance space. While traditional feature selection algorithms have the goal of finding the best feature subset that is relevant to both the learning task and the selected inductive learning algorithm, the task of ensemble feature selection has the additional goal of finding a set of feature subsets that will promote disagreement among the base classifiers [9].

Feature selection algorithms, including ensemble feature selection, are typically composed of the following components [1,9]: (1) search strategy, that searches the space of feature subsets; and (2) fitness function, that inputs a feature subset and outputs a numeric evaluation. The search strategy's goal is to maximize this function.

In [12] we presented a technique for building ensembles of simple Bayesian classifiers in random subspaces. We considered also a hill-climbing-based refinement cycle, which improved the accuracy and diversity of the base classifiers built on random feature subsets. In [11] we considered an application of the technique to the problem of acute abdominal pain classification. The main conclusions in [11, 12] are that, in the fitness function guiding the search in ensemble feature selection, accuracy and diversity are both important, but the degree of importance differs with different data sets.

In this paper, our focus is on the search strategy for ensemble feature selection, rather than on the fitness function. We develop two new strategies, in addition to already known strategies, which include random subspacing, hill-climbing search, and genetic search. Our new strategies are sequential- or greedy-search-based, and often less time consuming than genetic search or hill climbing.

In Section 2 general issues with an ensemble of simple Bayesian classifiers and ensemble feature selection are considered. In Section 3 we present four search strategies for ensemble feature selection and in the next section experiments with these are discussed. We conclude briefly in Section 4 with a summary and further research topics.

2. Feature selection for ensembles of simple Bayesian classifiers

An ensemble of classifiers is created by using training data to build several base classifiers, which are applied in combination to derive a final classification [4]. To be effective, an ensemble should consist of high-accuracy classifiers that disagree on their predictions [2]. In this paper, ensembles are generated that comprise simple Bayesian classifiers each of which is separately formed from a training set, taking into account only the features of the corresponding selected feature subset.

Ho [6] has shown that simple random selection of feature subsets may be an effective technique for ensemble feature selection because the lack of accuracy in the ensemble members is compensated for by their diversity. Random subspacing is used as a base in a number of ensemble feature selection strategies, e.g. GEFS [9] and HC [3].

To measure the disagreement of a base classifier and the whole ensemble, we calculate the diversity of the base classifier over the instances of the validation set as an average difference in classifications of all possible pairs of classifiers including the given one [12].

In this research, for each feature subset, we calculate a goodness measure using the fitness function proposed by Opitz [9]. The fitness $Fitness_i$ of a classifier corresponding to a feature subset is proportional to the classification accuracy acc_i and the diversity div_i of the classifier:

$$Fitness_i = acc_i + \alpha \cdot div_i \,, \qquad (1)$$

where α is the coefficient of the degree of the influence of diversity. When class distribution is uneven, accuracy in (1) should be replaced with the average of sensitivity and specificity, as we do in this research.

There are two major approaches applied in forming the *method for integration* in ensembles $F(y_1, \ldots, y_S)$: (1) the *combination* approach, where the base classifiers produce their classifications and the final result is composed of these; and (2) the *selection* approach, where one of the classifiers is selected and the final result is the result produced by it. For both approaches, there are static and dynamic methods [10]. In contrast to the static methods, the integration procedure of the dynamic methods depends on each instance being processed.

In our experiments we use five different integration methods: (1) cross-validation majority (a static selection approach, SS) [8]; (2) weighted voting (WV) [2] (a static combination approach); (3) dynamic selection (DS) [10] (a dynamic selection approach); (4) dynamic voting (DV) [10] (a dynamic combination approach); and (5) dynamic voting with selection (DVS) [12] (a dynamic hybrid approach). The three dynamic approaches are based on the same local accuracy estimates obtained using the weighted nearest neighbor prediction.

3. Search strategies for feature subset selection in ensembles

In this section, we consider four different search strategies for ensemble feature selection: (1) Hill Climbing (HC); (2) Genetic Ensemble Feature Selection (GEFS); (3) Ensemble Forward Sequential Selection (EFSS); and (4) Ensemble Backward Sequential Selection (EBSS).

The use of a hill-climbing search as a local-search wrapper-like approach has been shown to be effective for a single feature subset selection [8]. The Hill Climbing (HC) ensemble feature selection strategy, which we use in this research, proposed in [3], is composed of two major phases: (1) construction of the initial ensemble in random subspaces; and (2) iterative refinement of the ensemble members with sequential mutation hill climbing. Initial feature subsets are constructed using the random subspace method. Then, the initial ensemble is formed. Further, an iterative refinement of the ensemble members is used to improve the accuracy and diversity of the base classifiers. The iterative refinement is based on a hill-climbing search. For all the feature subsets, an attempt is made to switch (include or delete) each feature. If the resulting feature subset produces better performance on the validation set, that change is kept. This process is continued until no further improvements are possible.

The use of genetic search has also been an important direction in the feature selection research. Genetic algorithms have been shown to be effective global optimization techniques in feature subset selection. The use of genetic algorithms for ensemble feature selection was first proposed in [9]. The Genetic Ensemble Feature Selection (GEFS) strategy [9] begins, as HC, with creating an initial population of classifiers where each classifier is generated by randomly selecting a different subset of features. Then, new candidate classifiers are continually produced by using the genetic operators of crossover and mutation on the feature subsets. After a number of generations, the fittest individuals make up the population which comprises the ensemble [9]. In our implementation, the representation of each individual (a feature subset) is simply a constant-length string of bits, where each bit corresponds to a particular feature. The crossover operator uses uniform crossover, in which each feature of the two children takes randomly a value from one of the parents. The feature subsets of two individuals in the current population are chosen proportional to fitness. The mutation operator randomly toggles a percentage of bits in an individual.

Besides, in this paper we propose two new ensemble feature selection strategies, EFSS and EBSS. These are sequential feature selection strategies, which add or subtract features using a hill-climbing procedure, and have polynomial complexity. The most frequently studied variants of plain sequential feature selections algorithms (which select a single feature subset) are forward and backward sequential selection, FSS and BSS [1]. FSS begins with zero attributes, evaluates all feature subsets with exactly one feature, and selects the one with the best performance. It then adds to this subset the feature that yields the best performance for subsets of the next larger size. The cycle repeats until no improvement is obtained from extending the current subset. BSS instead begins with all features and repeatedly removes a feature whose removal yields the maximal performance improvement [1]. EFSS and EBSS iteratively apply FSS or BSS to form each of the base classifiers using a predefined fitness function.

EFSS and EBSS have polynomial complexity with regard to the number of features: $O(S \cdot N \cdot N')$, where S is the number of base classifiers, N is the total number of features, and N' is the number of features included or deleted on average in an FSS or BSS search. HC has similar polynomial complexity $O(S \cdot N \cdot N_{passes})$, where N_{passes} is the average number of passes through the feature subsets in HC until there is some improvement (usually no more than 4). The complexity of GEFS does not depend on the number of features, and is $O(S' \cdot N_{gen})$, where S' is the number of individuals (feature subsets) in one generation, and N_{gen} is the number of generations.

4. Experiments

The experiments were conducted on three large data sets with cases of acute abdominal pain (AAP): (1) Small-AAP I; (2) Medium-AAP II; and (3) Large-AAP III, with the numbers of instances respectively 1254, 2286, and 4020 [13]. These data sets represent the same problem of separating acute appendicitis from other diseases that cause acute abdominal pain. Each data set includes 18 features from history-taking and clinical examination [13].

For each data set, for the sake of performance comparison, we used the same division into training and test sets, as in [13]. We used the reduced training data sets with approximate ratio of ½ of instances classified as appendicitis and the other half classified as other diagnoses. Additionally, 20 percent instances from the original training sets were transferred to the validation sets using random sampling. 30 test runs were made for 30 train/validation splits of the original training sets for each of the four algorithms. We experimented with six different values of the diversity coefficient α: 0, 0.25, 0.5, 1, 2, and 4. The size of ensemble was selected to be equal to 25.

At each run of the algorithm, we collected accuracies for the five types of the integration of classifiers: SS, WV, DS, DV, and DVS. In the dynamic integration methods, the number of nearest neighbors for local accuracy estimates was pre-selected from a set of six values: 1, 3, 7, 15, 31, 63, for each data set. Besides the classification accuracies of the base classifiers and the ensemble, and corresponding sensitivity and specificity values, we collected such characteristics as total ensemble diversity, ensemble coverage, and the average relative number of features in the base classifiers. All these characteristics were averaged over the 30 runs.

The test environment was implemented within the MLC++ framework (the machine learning library in C++) [7]. For the simple Bayesian classifier, the numeric features were discretized into ten equal-length intervals (or one per observed value, whichever was less).

Parameter settings for the genetic search in GEFS include a mutation rate of 50% (as proposed in [9]), a population size of 25, a search length of 100 feature subsets, of which 50 are offsprings of the current population of 25 classifiers generated with the crossover operator, and 25 are mutated offsprings. Only one generation of individuals was produced, as our pilot experiments have shown that creating more generations does not increase performance, and sometimes even decreases it by 2-3%; this is probably due to overfitting the training data.

In Table 1, the experimental results for the four search strategies on the three data sets are presented. The table includes the name of a data set, the best selected α (α), the average of sensitivity and specificity for the five integration methods (*SS*, *WV*, *DS*, *DV*, *DVS*) for the four search strategies (HC, GEFS, EFSS and EBSS), the average of sensitivity and specificity of the simple Bayes on the whole feature set (*Bayes*), the average relative number of features selected (*feat*), and the improvement of the final ensemble in comparison with the random subspace ensemble with regard to the average of sensitivity and specificity (*impr*). The best results for each search strategy are given in italic type, and for the whole data set – in bold type.

Table 1. Results for the four search strategies

data set	strategy	α	SS	WV	DS	DV	DVS	Bayes	feat	Impr
Small (AAP I)	HC	2	0.705	0.753	0.740	0.757	*0.767*	0.747	0.312	0.022
	GEFS	¼	0.741	*0.761*	0.743	*0.761*	0.752		0.559	0.016
	EFSS	4	0.738	0.747	0.745	0.766	***0.768***		0.093	***0.023***
	EBSS	4	0.733	0.749	0.731	0.753	*0.757*		0.428	0.012
Medium (AAP II)	HC	4	0.553	0.572	*0.599*	0.585	0.596	0.516	0.240	0.036
	GEFS	4	0.576	0.547	0.589	0.577	*0.590*		0.195	0.027
	EFSS	2	0.579	0.570	***0.601***	0.583	0.589		0.133	***0.038***
	EBSS	¼	0.540	*0.579*	0.565	*0.579*	0.569		0.591	0.016
Large (AAP III)	HC	4	0.816	0.806	0.836	0.854	*0.864*	0.828	0.183	0.025
	GEFS	1	0.825	0.823	0.827	0.834	*0.843*		0.343	0.004
	EFSS	4	0.824	0.818	0.854	0.863	***0.870***		0.078	***0.031***
	EBSS	4	0.833	0.828	0.838	0.854	*0.862*		0.244	0.023

From Table 1, one can see that for all of the three data sets, most ensembles perform significantly better than the single global simple Bayes. For the data sets AAP I and AAP III, the best integration technique is DVS, and for the data set AAP II the best integration technique is DS, which give significantly better results than the single simple Bayes (the statistical significance is checked with the 1-tailed Student *t*-test with 0.95 level of significance). An interesting finding is that the best search strategy is EFSS for every data set. The average of sensitivity and specificity in these cases rivals the best previously published results for these data sets [11, 13]. The good performance of EFSS can be explained by the fact that EFSS is able to generate classifiers with better diversity, starting with zero feature subsets, in comparison with EBSS and the other strategies. EFSS generates also extremely compact base classifiers, including from 9 to 13% of features on average (less than 3 features). Dynamic integration is in general much better than static integration for these data sets, better utilizing the diversity of the base classifiers, supporting the results presented in [11, 12]. The selected values of α are different for different search strategies, which means that the ensemble diversity is important, as was shown also in [11, 12], but the degree of importance depends on the search strategy used, and not only on the data set.

5. Conclusion

In this paper we considered four search strategies for ensemble feature selection, two of which were new sequential search strategies, EFSS and EBSS. We conducted a number of experiments on a collection of data sets from the medical field of acute appendicitis. In many cases the ensembles of simple Bayesian classifiers had higher performance than the single global simple Bayesian classifier. The best search strategy was EFSS, generating more diverse ensembles with more compact base classifiers. The average of sensitivity and specificity of EFSS rivaled the best previously published results. The proposed search algorithms could be useful for other medical domains, especially including many features with complex inter-feature dependencies.

In future research, it would be interesting to consider other search strategies, such as beam search and simulated annealing, and to try to find a better configuration for GEFS, as the present results with the genetic search were disappointing, not being improved with generations. Another interesting topic for further research is the check of the presented findings on other data sets with different characteristics.

Acknowledgments: We would like to thank Science Foundation Ireland for financial support. This research is partly supported by the COMAS Graduate School of the University of Jyväskylä, Finland. We would like to thank the machine learning library in C++ for the source code used in this study. The Acute Abdominal Pain (AAP) data sets were kindly provided by the Laboratory for System Design, Faculty of Electrical Engineering and Computer Science, University of Maribor, Slovenia and the Theoretical Surgery Unit, Department of General and Trauma Surgery, Heinrich-Heine University, Düsseldorf, Germany.

6. References

[1] D.W. Aha, and R.L. Bankert, “A Comparative Evaluation of Sequential Feature Selection Algorithms”, In: D. Fisher and H. Lenz (eds.), Proc. 5th Int. Workshop on Artificial Intelligence and Statistics, 1995, pp. 1-7.

[2] E. Bauer, and R. Kohavi, “An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants”, Machine Learning, Vol. 36, Nos. 1,2, 1999, pp. 105-139.

[3] P. Cunningham, and J. Carney, “Diversity Versus Quality in Classification Ensembles Based on Feature Selection”, In: R.L.deMántaras & E.Plaza (eds.), Proc. ECML 2000, 11th European Conf. On Machine Learning, Barcelona, Spain, LNCS 1810, Springer, 2000, pp. 109-116.

[4] T. G. Dietterich, “Ensemble Learning Methods”, In: M.A. Arbib (ed.), Handbook of Brain Theory and Neural Networks, 2nd ed., MIT Press, 2001.

[5] Fayyad, U., G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Advances in Knowledge Discovery and Data Mining, AAAI/ MIT Press, 1997.

[6] T. K. Ho, “The Random Subspace Method for Constructing Decision Forests”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 8, 1998, pp. 832-844.

[7] R. Kohavi, and D. Sommerfield, J. Dougherty, “Data Mining Using MLC++: A Machine Learning Library in C++”, Tools with Artificial Intelligence, IEEE CS Press, 1996, pp. 234-245.

[8] Kohavi, R., Wrappers for Performance Enhancement and Oblivious Decision Graphs, Dept. of Computer Science, Stanford University, Stanford, USA, PhD Thesis, 1995.

[9] D. Opitz, “Feature Selection for Ensembles”, In: Proc. 16th National Conf. on Artificial Intelligence, AAAI, 1999, pp. 379-384.

[10] S. Puuronen, V. Terziyan, and A. Tsymbal, “A Dynamic Integration Algorithm for an Ensemble of Classifiers”, In: Z.W. Ras, A. Skowron (eds.), Foundations of Intelligent Systems: ISMIS’99, LNAI, Vol. 1609, Springer, 1999, pp. 592-600.

[11] A. Tsymbal, and S. Puuronen, “Ensemble Feature Selection with the Simple Bayesian Classification in Medical Diagnostics”, In: Proc. 15th IEEE Symp. on Computer-Based Medical Systems CBMS’2002, Maribor, Slovenia, IEEE CS Press, 2002, pp. 225-230.

[12] A. Tsymbal, S. Puuronen, and D. Patterson, “Feature Selection for Ensembles of Simple Bayesian Classifiers”, In: Foundations of Intelligent Systems: ISMIS’2002, LNAI, Vol. 2366, Springer, 2002, pp. 592-600.

[13] M. Zorman, H.-P. Eich, P. Kokol, and C. Ohmann, “Comparison of Three Databases with a Decision Tree Approach in the Medical Field of Acute Appendicitis”, In: V.Patel et al. (eds.), Proc. 10th World Congress on Health and Medical Informatics Medinfo’2001, Vol.2, London, UK, IOS Press, 2001, pp. 1414-1418.