Neural Comput & Applic (2018) 29:447–457
https://doi.org/10.1007/s00521-016-2458-6

ORIGINAL ARTICLE

Prototype selection for dynamic classifier and ensemble selection

Rafael M. O. Cruz[1] · Robert Sabourin[1] · George D. C. Cavalcanti[2]

Received: 4 March 2016 / Accepted: 1 July 2016 / Published online: 14 July 2016

Abstract In dynamic ensemble selection (DES) techniques, only the most competent classifiers, for the classification of a specific test sample, are selected to predict the sample's class labels. The key in DES techniques is estimating the competence of the base classifiers for the classification of each specific test sample. The classifiers' competence is usually estimated according to a given criterion, which is computed over the neighborhood of the test sample defined on the validation data, called the region of competence. A problem arises when there is a high degree of noise in the validation data, causing the samples belonging to the region of competence to not represent the query sample. In such cases, the dynamic selection technique might select the base classifier that overfitted the local region rather than the one with the best generalization performance. In this paper, we propose two modifications in order to improve the generalization performance of any DES technique. First, a prototype selection technique is applied over the validation data to reduce the amount of overlap between the classes, producing smoother decision borders. During generalization, a local adaptive K-Nearest Neighbor algorithm is used to minimize the influence of noisy samples in the region of competence. Thus, DES techniques can better estimate the classifiers' competence. Experiments are conducted using 10 state-of-the-art DES techniques over 30 classification problems. The results demonstrate that the proposed scheme significantly improves the classification accuracy of dynamic selection techniques.

Keywords Ensemble of classifiers · Dynamic ensemble selection · Prototype selection

✉ Rafael M. O. Cruz
cruz@livia.etsmtl.ca

Robert Sabourin
Robert.Sabourin@etsmtl.ca

George D. C. Cavalcanti
gdcc@cin.ufpe.br

1 Département de génie de la production automatisée, École de Technologie Supérieure (ETS), Montreal, QC, Canada

2 Centro de Informática, Universidade Federal de Pernambuco, Recife, Brazil

1 Introduction

In the last few years, dynamic ensemble selection (DES) [3] has become an active research topic in multiple classifier systems. The rationale behind such techniques resides in the observation that not every classifier in the pool is an expert in classifying all unknown samples. Each base classifier[1] is an expert in a different local region of the feature space [32].

Dynamic selection techniques consist, based on a pool of classifiers C, in finding a single classifier c_i, or an ensemble of classifiers C', that has (or have) the most competent classifiers to predict the label for a specific test sample, $\mathbf{x}_j$. The most important component of DES techniques is how the competence level of the base classifier is measured, given a specific test sample $\mathbf{x}_j$. Usually, the competence of a base classifier is estimated based on instances that are similar to the query instance, using the K-Nearest Neighbors (KNN) technique and a set of labeled samples, which can be either the training or validation set. In this paper, we refer to such a set as the dynamic selection dataset (DSEL), following the conventions of the dynamic selection literature [3, 9]. The set with the K-

1 The term base classifier refers to a single classifier belonging to an ensemble or a pool of classifiers.

Nearest Neighbors of a given test sample $\mathbf{x}_j$ is called the region of competence and is denoted by $\theta_j = \{\mathbf{x}_1, \ldots, \mathbf{x}_K\}$. The samples belonging to θ_j are used to estimate the competence of the base classifiers, for the classification of $\mathbf{x}_j$, based on various criteria, such as the overall accuracy of the base classifier in this region [31], ranking [20], ambiguity [12], oracle [17] and probabilistic models [30].

A problem arises with dynamic selection techniques when the samples in the local region are not representative enough of the query sample. This may be seen in cases in which a high degree of overlap is present between the classes, and as a result of noise or outliers. As reported in [7], the performance of dynamic selection techniques is very sensitive to the distribution of DSEL.

In order to illustrate how the presence of noise in DSEL can lead to poor classification results by using a dynamic selection technique, we perform a case study using the synthetic P2 problem proposed in [25]. The P2 is a bi-dimensional two-class synthetic classification problem in which each class is defined in multiple decision regions delimited by polynomial and trigonometric functions.

For this example, the META-DES framework proposed in [9] is considered since it outperformed several dynamic selection techniques in multiple classification benchmarks. The P2 problem was generated using the same methodology reported in [7]: 500 samples for the training set ($\mathcal{T}$), 500 instances for the dynamic selection dataset, DSEL, and 2000 samples for the test set, $\mathcal{G}$. The original distribution of DSEL is shown in Fig. 1a. The red circle and blue cross represent samples belonging to class 1 and class 2, respectively.

Since there is no overlap between the classes in the original distribution, we generate noise in DSEL by switching the labels of samples that are close to the decision border with a 25 % probability (Fig. 1b). Samples that had their class labels changed are highlighted in green. Figure 1c, d show the approximation of the P2 border achieved by the META-DES framework plotted over the test distribution. Figure 1c presents the decision achieved using the original distribution of DSEL. In contrast, Fig. 1d presents the decision obtained using DSEL with 25 % of added noise. We can observe that the META-DES fails to obtain a good approximation of the decision boundary of the P2 Problem when noise is added to DSEL. Moreover, the errors committed by the META-DES occur in regions of the feature space where the presence of noise in DSEL is more evident.

This work therefore aims to improve the classification accuracy of dynamic selection techniques by reducing the presence of noise in DSEL. The proposed scheme is based on two steps: The first modification proposed in this paper applies a prototype selection mechanism to the dynamic selection set, DSEL, in order to eliminate instances highly likely to be noise, and also reduces the amount of overlap between the classes. The Edited Nearest Neighbor (ENN) [27] rule is used for this purpose. Secondly, the local regions of the query sample are estimated using an adaptive KNN rule (AKNN), which shifts the region of competence from the class border to the class centers. Samples that are more likely to be noise are less likely to be selected to compose the region of competence. As such, we expect the dynamic selection technique to be able to better estimate the competence level of a base classifier, leading to better generalization performance. It should be mentioned that the proposed method can be applied to any dynamic selection technique that uses local information in estimating the competence of the base classifier.

The proposed approach is evaluated using 10 state-of-the-art dynamic classifier and ensemble selection techniques over 30 classification datasets. We evaluate four scenarios: (I) The dynamic selection techniques using the original dynamic selection dataset and the standard KNN algorithm for computing the region of competence θ_j; (II) the ENN is applied to edit DSEL and the standard KNN is used; (III) only the AKNN technique is used, and (IV) both the ENN and the AKNN techniques are used. The following research questions are analyzed: (1) Does the prototype selection technique lead to an improvement in classification accuracy? (2) Which scenario produces the best recognition rates? (3) Which dynamic selection technique benefits the most from the proposed scheme?

This paper is organized as follows: The proposed approach is detailed in Sect. 2. The experimental study is conducted in Sect. 3. Finally, our conclusion and future works are presented in the last section.

2 Proposed method

Two changes are proposed in this paper; one during the training stage, and the other in the generalization stage. In the training stage, we apply a prototype selection technique in the dataset DSEL in order to remove noise and outliers. To that end, the Edited Nearest Neighbor technique is considered since it is able to significantly reduce the presence of noise in the dataset, thereby improving the KNN performance [27]. During the generalization stage, given a new test sample $\mathbf{x}_{j,test}$, the region of competence θ_j is computed based on the samples in the edited dynamic selection dataset, denoted by DSEL$'$, using a local adaptive distance rule. Both techniques are presented in the following sections.

2.1 Edited nearest neighbor (ENN)

There are three types of prototype selection mechanisms available [14]: condensation, edition and hybrid.

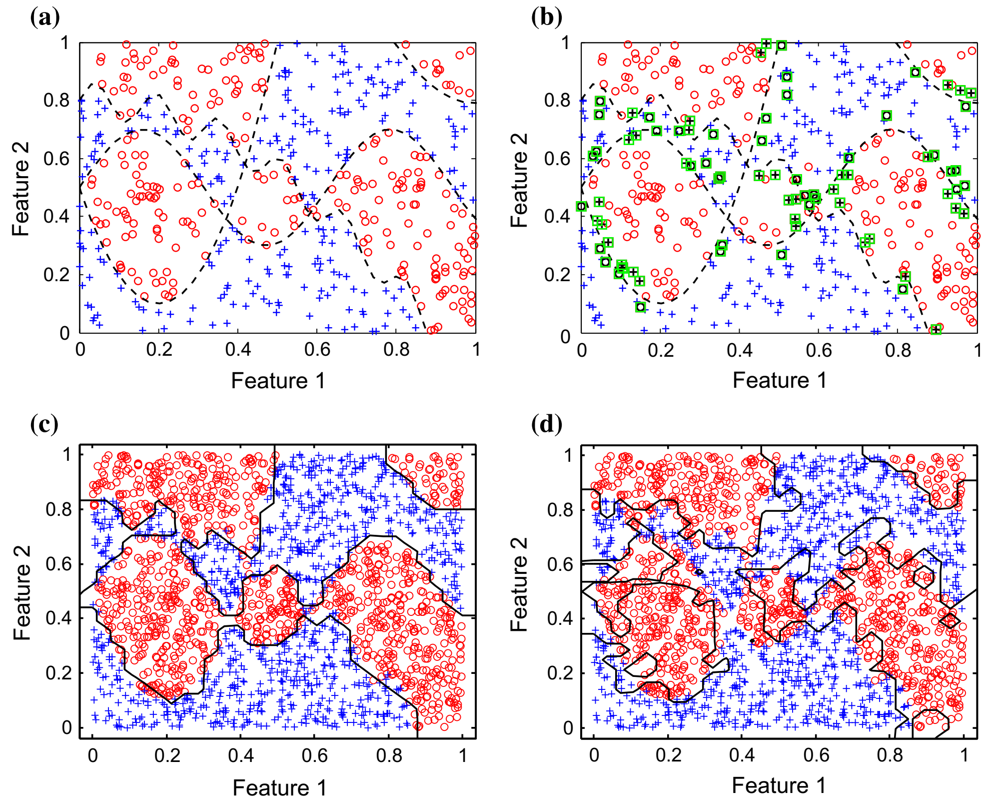

Fig. 1 Case study using the synthetic P2 problem. The *red circle* represents the class 1 and the *blue cross* the class 2. The axes represent the values of the two features of the P2 problem. **a** The original distribution of DSEL. **b** The distribution of DSEL with 25 % of added noise by switching labels of samples close to the class borders. The noisy samples are *highlighted in green*. **c** Result of the META-DES framework using the Original DSEL. **d** Results of the META-DES framework using the noisy DSEL (color figure online)

Condensation techniques are used in order to reduce the dataset size, without losing the generalization performance of the system. Edition techniques aim to improve the performance of the KNN algorithm by removing instances with a high risk of being noise. The editing process occurs in regions of the feature space with a high degree of overlap between classes, producing smoother class boundaries. Hybrid techniques perform both a condensation of the data and edition of the class borders. Since our goal is to improve the classification accuracies, an edition technique is performed. The Edited Nearest Neighbor (ENN) [27] is used since it is a very well-known technique for removing noise and decreasing the amount of overlap in the class borders, producing smoother decision borders. Moreover, the ENN technique is known to significantly improve the performance of the KNN [14].

Algorithm 1 The Edited Nearest Neighbor rule

Input: Dynamic Selection Dataset DSEL

DSEL$'$ = DSEL
for each $\mathbf{x}_{j,DSEL} \in$ DSEL **do**
  if $label(\mathbf{x}_{j,DSEL}) \neq label(KNN(\mathbf{x}_{j,DSEL}))$ **then**
    DSEL$'$ = DSEL$'$ $\setminus \{\mathbf{x}_{j,DSEL}\}$
  end if
end for
return DSEL$'$

Given the dynamic selection dataset DSEL, the ENN algorithm works as follows (Algorithm 1): For each instance $\mathbf{x}_{j,DSEL} \in$ DSEL, the class label of $\mathbf{x}_{j,DSEL}$ is predicted using the KNN algorithm using a leave-one-out procedure. A $K = 3$ was used, as suggested by Wilson [27], in order to satisfy the asymptotic properties of the NN technique. If $\mathbf{x}_{j,DSEL}$ is misclassified by the KNN technique, it is removed from the set, since $\mathbf{x}_{j,DSEL}$ is in a region of the feature space where the majority of samples belongs to a different class. The edited dynamic selection dataset, denoted by DSEL$'$, is obtained at the end of the process.

It should be mentioned that the ENN does not remove all samples in the class borders and that the intrinsic geometry of the class borders and the distribution of the classes are preserved. Only instances that are associated with a high degree of instance hardness, i.e., those for which the majority of neighbors belong to a different class, are removed. As reported in [21], these samples have a reputation for being hard to be correctly classified by the majority of learning algorithms. Only the classifiers that overfitted the training data are able to predict its correct class label. In such cases, the dynamic selection technique might select the base classifier that overfitted the local region rather than the one that has the best generalization performance in the region. By removing these instances, we expect the dynamic selection techniques to be able to better estimate the base classifier's competences.

2.2 *K*-nearest neighbor with local adaptive distance

The locally adaptive distance for KNN was proposed in [26]. For each sample $\mathbf{x}_{j,DSEL'}$ in the edited dynamic selection dataset DSEL$'$, the largest hypersphere centered on $\mathbf{x}_{j,DSEL'}$, which excludes all samples in DSEL$'$ with a different class label, is constructed (Fig. 2). Such a hypersphere is built by computing its radius $R_{j,DSEL'}$, which is measured as the minimum distance between the sample $R_{j,DSEL'}$ and a sample from a different class $\mathbf{x}_{jk,DSEL'}$ (Eq. 1):

$$R_{j,DSEL'} = d\left(\mathbf{x}_{j,DSEL'}, \mathbf{x}_{jk,DSEL'}\right) - \epsilon \tag{1}$$

where $d\left(\mathbf{x}_{j,DSEL'}, \mathbf{x}_{jk,DSEL'}\right)$ is the Euclidean distance between the instances $\mathbf{x}_{j,DSEL'}$ and $\mathbf{x}_{jk,DSEL'}$, ϵ is a small number (in this work $\epsilon = 0.01$). $w_{j,DSEL'}$ and $w_{jk,DSEL'}$ are the labels of $\mathbf{x}_{j,DSEL'}$ and $\mathbf{x}_{jk,DSEL'}$, respectively.

Each instance belonging to DSEL$'$ is associated with a hypersphere of radius $R_{j,DSEL'}$. The hypersphere associated with each sample delimits the region within which its class label can be generalized to other samples without making an error [26]. The hypersphere associated with samples that are closer to the class center have a larger radius since they are more distant from samples from different classes, when compared to those hyperspheres that are associated with samples that are closer to the class boundaries. Figure 2 illustrates an example of the hypersphere associated with different samples from the P2 problem.

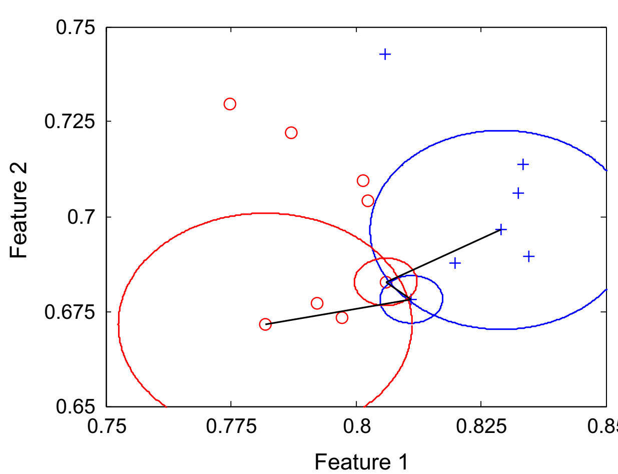

Fig. 2 Example of the hypersphere associated with the samples in DSEL$'$, considering the P2 problem. The *red circle* and the *blue cross* represent samples belonging to class 1 and class 2, respectively. The *X*- and *Y*-axes indicate the values of the two features of the P2 problem (color figure online)

The adaptive distance between a given test sample $\mathbf{x}_{j,test}$ and a sample belonging to $DSEL'$, $\mathbf{x}_{j,DSEL'}$, is obtained using Eq. 2.

$$d_{\text{adaptive}}(\mathbf{x}_{j,test}, \mathbf{x}_{j,DSEL'}) = \frac{d(\mathbf{x}_{j,\text{test}}, \mathbf{x}_{j,DSEL'})}{R_{j,DSEL'}} \tag{2}$$

The distance is said to be adaptive since the influence of each sample is normalized by a factor $R_{j,DSEL'}$, which changes according to the spatial location of each instance in DSEL$'$. The larger the value of $R_{j,DSEL'}$ (i.e., larger hypersphere), the lower the value of d_{adaptive}. The A-KNN technique is beneficial in regions where there is a high degree of overlap between the two classes, since it tends to identify samples that have larger hyperspheres as the nearest neighbors to the query sample. As reported in [26], the majority of *K*-Nearest Neighbors selected are more likely to have the same class label as the query sample. Thus, the dynamic selection algorithm can better estimate the competence of the base classifiers for the classification of $\mathbf{x}_{j,test}$.

2.3 Case study

Using the same distributions of the P2 problem discussed in Sect. 1, if we apply the ENN technique in editing the dynamic selection dataset, the overlap in the decision boundary is significantly removed. Figure 3a shows the distribution of the edited dynamic selection dataset $DSEL'$ using the ENN prototype selection technique. Noisy samples are highlighted in green. We can see that the majority of noisy samples were removed from DSEL. In addition, we can see that the geometry of the decision border is still preserved. Figure 3b shows the result of the META-DES technique using the $DSEL'$ in computing the local regions. The META-DES can have a closer approximation of the real decision boundary of the P2 problem. However, it can be seen that there are still some outliers in the edited DSEL, and their presence still negatively affects the performance of the system.

The adaptive distance comes in handy in those cases since there is no guarantee that the ENN will completely remove all noisy samples from DSEL. If we also use the adaptive distance (Fig. 3c) in computing the region of competence θ_j, the META-DES can obtain a decision boundary that is close to those obtained using a noise-free DSEL. Thus, by editing the dynamic selection dataset and the adaptive KNN distance, we can obtain a good approximation of the decision boundary of the P2 problem, even with a high noise presence.

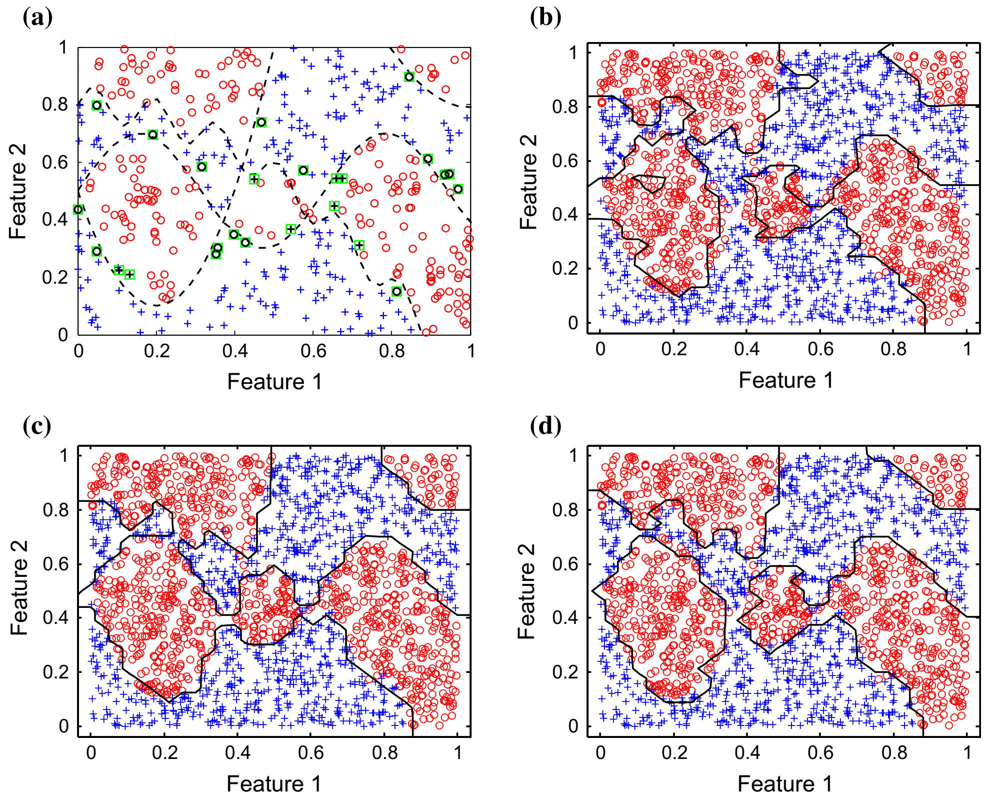

Fig. 3 Case study using the two-dimensional P2 problem. The axes represent the values of the two features of the P2 problem: **a** Distribution of DSEL after applying the ENN technique to clean the border. Noisy samples are *highlighted* in *green*; **b** Result of the META-DES framework using DSEL′ for computing the local regions; **c** Result of the META-DES using the adaptive distance (AKNN); **d** Result of the META-DES framework using both the ENN and the AKNN techniques (color figure online)

3 Experiments

In this section, we compare the impact of the adaptive distance and the editing of the class boundaries using several state-of-the-art dynamic classifier selection and dynamic ensemble selection techniques found in the literature.

3.1 Dynamic selection methods

A total of 10 dynamic selection techniques were considered in the experiments. In order to have a balance between dynamic classifier selection (DCS) and dynamic ensemble selection (DES), we considered five techniques from each paradigm. In addition, based on the dynamic selection taxonomy proposed in [3], there were five categories: Ranking, Accuracy, Oracle, Probabilistic and Behavior. To ensure the availability of a diverse set of techniques, we considered at least one technique taken from each category. We also included the META-DES in the experimental study which was published after the survey, and could be considered as belonging to a different category (meta-learning). Thus, methods that use different sources of information for estimating the competence level of the base classifiers were considered in the experimental study. Table 1 illustrates the 10 dynamic selection techniques considered in this work.

For dynamic classifier selection, the following techniques were considered: Local classifier Accuracy (LCA) [31], Overall Local Accuracy (OLA) [31], Modified Local Accuracy (MLA) [22], Classifier ranking (RANK) [20] and the Multiple Classifier Behavior (MCB) [15]. The following techniques for dynamic ensemble selection were considered: *K*-Nearest Oracles Eliminate (KNORA-E) [17], *K*-Nearest Oracles Union (KNORA-U) [6], Randomized Reference Classifier (DES-PRC) [28], *K*-Nearest Output Profiles (KNOP) [5, 6], and the META-DES framework [9]. The pseudo-code for each technique can be found in the following survey [3], and in [9], for the META-DES framework.

3.2 Datasets

The experiments were conducted on 30 datasets taken from five different data repositories. Sixteen datasets were taken from the UCI machine learning repository [2], four from the STATLOG project [16], four from the Knowledge Extraction based on Evolutionary Learning (KEEL) repository [1], four from the Ludmila Kuncheva Collection of real medical data [18], and two artificial datasets generated with the Matlab PRTOOLS toolbox [13]. The experimental study is focused on small size datasets, since, as reported by Cavalin et al. [6], dynamic selection techniques have been shown to be an effective tool for

Table 1 Dynamic selection techniques considered in the experiments

Technique	Category	Reference
DCS		
Classifier rank (RANK)	Ranking	Sabourin et al. [20]
Local Classifier Accuracy (LCA)	Accuracy	Woods et al. [31]
Overall Local Accuracy (OLA)	Accuracy	Woods et al. [31]
Modified Local Accuracy (MLA)	Accuracy	Smits [22]
Multiple Classifier Behavior (MCB)	Behavior	Giacinto et al. [15]
DES		
K-Nearests Oracles Eliminate (KNORA-E)	Oracle	Ko et al. [17]
K-Nearests Oracles Union (KNORA-U)	Oracle	Ko et al. [17]
Randomized Reference Classifier (RRC)	Probabilistic	Woloszynski et al. [29]
K-Nearests Output Profiles (KNOP)	Behavior	Cavalin et al. [6]
META-DES	Meta-Learning	Cruz et al. [9]

Pseudo-code for each technique can be found in the following survey [3], and in [9], for the META-DES framework

problems where the level of uncertainty for recognition is high due to few training samples being available. However, a few larger datasets, such as the Magic gamma telescope, phoneme and Adult, were also considered in order to evaluate the performance of the proposed scheme for different types of classification problems.

Since ensemble methods have recently become popular in dealing with the class imbalance problem [23, 11, 19], several imbalanced datasets, such as Ecoli, Glass, Satimage and Phoneme, were also considered. Table 2 presents the main characteristics of the 30 classification datasets. The imbalanced ratio (IR) is measured by the number of instances of the majority class per instance of the minority class. Thus, a higher IR value indicates a higher degree of imbalance.

In order to ensure a fair comparison between the results obtained by the proposed technique and those from the DES literature, the same experimental setup as in previous works [9] is considered. For each dataset, the experiments were carried out using 20 replications. For each replication, the datasets were randomly divided on the basis of 50 % for training, $\mathcal{T}$, 25 % for the dynamic selection dataset, *DSEL*, and 25 % for the generalization set, $\mathcal{G}$. The divisions were performed while maintaining the prior probabilities of each class. Since the META-DES framework requires an additional training step for the training of the meta-classifiers (meta-training), 25 % of the training set was used in the meta-training phase. The pool of classifiers C was composed of 100 Perceptrons generated using the Bagging technique. The size of the region of competence (neighborhood size) K was equally set at 7 for all techniques. The hyper-parameters for the META-DES framework were set according to guidelines proposed by the authors [7, 8].

3.3 Comparison between different scenarios

We evaluated four different scenarios for the dynamic selection techniques (Table 3).

For each scenario, we evaluated each dynamic selection technique over the 30 datasets, for a total of 300 experiments (30 datasets $\times$ 10 techniques) per scenario. To compare the four approaches, the Friedman rank analysis was conducted since it is a robust statistical method for comparing multiple techniques over several datasets [10].

For each dataset and dynamic selection method, the Friedman test ranks each scenario, with the best performing one getting rank 1, the second best rank 2, and so forth. Then, the average rank of each scenario is calculated. The best scenario is the one that obtained the lowest average rank. After the average ranks were computed, the post hoc Bonferroni–Dunn test was conducted for a pairwise comparison between the ranks achieved by each scenario. The performance of two techniques is significantly different if their difference in average rank is higher than the critical difference (CD) calculated by the Bonferroni–Dunn post hoc test. The average ranks of the four scenarios, as well as the results of the post hoc test, are presented using the CD diagram [10] (Fig. 4). We can see, based on the CD diagram, that the performance of Scenario IV is statistically better when compared to the other scenarios.

In addition to the Friedman analysis, we also conducted a pairwise comparison between Scenario I (without using the ENN and A-KNN) and the other test scenarios, using the sign test [10] calculated on the computed wins, ties and losses. The null hypothesis H_0 meant that both approaches yielded equivalent results, and a rejection in H_0 meant that the proposed approach was significantly better at a predefined significance level. In this work, we use the significance level $\alpha = 0.05$. To reject H_0, the number of wins

Table 2 Summary of the 30 datasets used in the experiments (Adapted from [9])

Database	No. of instances	Dimensionality	No. of classes	IR	Source
Pima	768	8	2	1.87	UCI
Liver disorders	345	6	2	1.37	UCI
Breast (WDBC)	568	30	2	1.86	UCI
Blood transfusion	748	4	2	3.20	UCI
Banana	1000	2	2	1.00	PRTOOLS
Vehicle	846	18	4	1.09	STATLOG
Lithuanian	1000	2	2	1.00	PRTOOLS
Sonar	208	60	2	1.14	UCI
Ionosphere	315	34	2	1.78	UCI
Wine	178	13	3	1.47	UCI
Haberman's survival	306	3	2	2.78	UCI
Cardiotocography (CTG)	2126	21	3	9.40	UCI
Vertebral column	310	6	2	2.1	UCI
Steel plate faults	1941	27	7	14.05	UCI
WDG V1	5000	21	3	1.02	UCI
Ecoli	336	7	8	71.50	UCI
Glass	214	9	6	8.44	UCI
ILPD	583	10	2	2.49	UCI
Adult	48,842	14	2	3.17	UCI
Weaning	302	17	2	1.00	LKC
Laryngeal1	213	16	2	1.62	LKC
Laryngeal3	353	16	3	4.19	LKC
Thyroid	215	5	3	12.05	LKC
German credit	1000	20	2	2.33	STATLOG
Heart	270	13	2	1.25	STATLOG
Satimage	6435	19	7	9.29	STATLOG
Phoneme	5404	6	2	2.41	ELENA
Monk2	4322	6	2	1.11	KEEL
Mammographic	961	5	2	1.05	KEEL
MAGIC gamma telescope	19,020	10	2	1.84	KEEL

The imbalanced ratio (IR) is measured by the number of instances of the majority class per instance of the minority class

Table 3 Four test scenarios

Scenario	ENN	Adaptive KNN
I	No	No
II	Yes	No
III	No	Yes
IV	Yes	Yes

needs to be greater than or equal to the critical value n_c calculated using Eq. 3:

$$n_c = \frac{n_{exp}}{2} + z_\alpha \frac{\sqrt{n_{exp}}}{2} \tag{3}$$

where n_{exp} is the total number of experiments (10 techniques $\times$ 30 datasets $=$ 300), and $z_\alpha = 1.645$, for a significance level of $\alpha = 0.05$. Hence, $n_c = 170.14$.

Considering Scenario IV, the number of wins, ties and losses are 195, 23 and 82, respectively. However, for computing the test, half the ties are added to the wins and the other half to the losses, which gives us 206.5 wins and 93.5 losses. H_0 is rejected since $206.5 > 170.14$. Scenario II also presented a significant gain in performance, with 186 wins and 114 losses, while the performance of Scenario III was statistically equivalent (152 wins and 148 losses).

Based on the statistical analysis, we can conclude that Scenarios II and IV achieve results that are statistically better when compared to Scenario I. Thus, the proposed scheme does indeed lead to significant gains in performance for dynamic selection techniques. We can also observe that the editing of DSEL using the ENN technique is the main factor in improving the classification performance, since Scenario II also presented a significant gain in

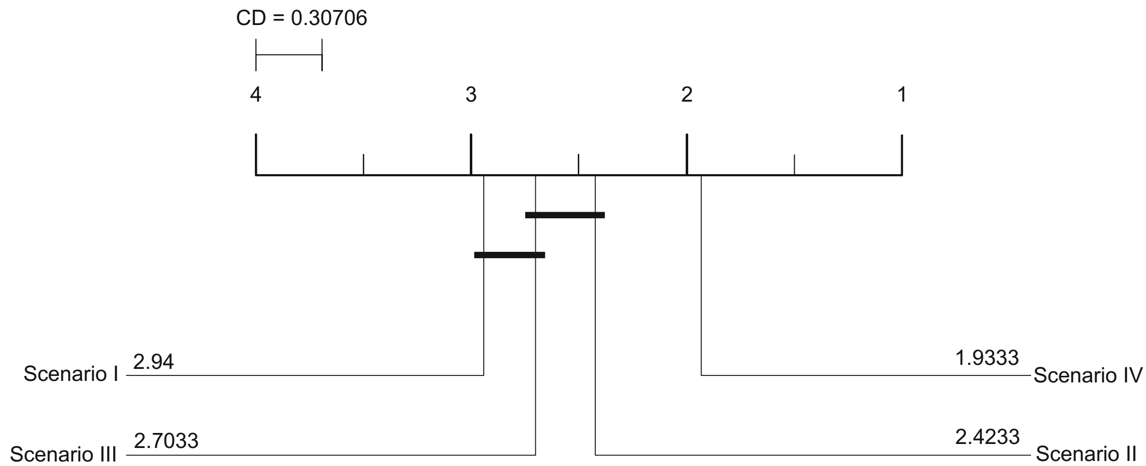

Fig. 4 Critical difference diagram considering the four test scenarios. The best algorithm is the one presenting the lowest average rank. Techniques that are statistically equivalent are connected by a *black bar*

performance when compared to Scenario I, while the performance of Scenario I and III was statistically equivalent.

3.4 Comparison between DES techniques

In order to identify which dynamic selection technique benefited the most from the proposed scheme, we conducted an analysis considering each technique separately. We performed a pairwise comparison between each DES technique using Scenarios I and IV. Only Scenario IV is considered in this analysis since it outperformed Scenarios II and III in the previous experiment. The comparison was conducted using the sign test calculated on the computed wins, ties and losses. The null hypothesis H_0 meant that the corresponding DES technique achieved equivalent results using Scenarios I and IV. In this case, the total number of experiments for each DES technique is equal to the number of datasets $n_{exp} = 30$.

In order to reject H_0 at $\alpha = 0.05$, the number of wins plus half the number of ties achieved by a dynamic selection technique must be greater than or equal to the critical value, $n_c = 19.5$. As shown in Fig. 5, the META-DES, OLA, LCA, KNORA-E, DCS-RANK and DES-PRC achieved significant performance gains using the proposed approach.

Furthermore, the Friedman test was used in order to compare the results of all the DES techniques over the 30 classification datasets (Fig. 6), using Scenarios I and IV. Techniques marked with an * are the ones using the ENN and A-KNN (Scenario IV). It can be seen that all DES techniques presented a lower average rank when using the proposed scheme (Scenario IV). Moreover, the techniques that are based purely on local accuracy information, such as LCA and OLA and DCS-RANK, presented a greater benefit, i.e., difference between the average ranks. For instance, the LCA* achieved an average rank of 9.96, while the average rank for the original LCA technique was 12.96. Techniques that are not based on the information extracted from the feature space, such as the MCB, which estimates the competence of the base classifier using the decision space, are the ones with smaller differences in average ranks (12.0 obtained by MCB against 11.4 achieved by the MCB*), which may simply be explained by the fact the ENN technique reduces the amount of overlap in the feature space rather than the decision space. Since the META-DES technique obtained the lowest average rank, we also present the classification accuracies obtained by the META-DES and META-DES* for the 30 classification datasets (Table 4). The best results are highlighted in bold.

3.5 Discussion

Looking at the classification results in Table 4, we can see that the proposed scheme works well when dealing with problems with few classes, even when considering datasets with a high degree of overlap between them, such as the Liver, Blood and Monk2, datasets. The proposed scheme failed to improve the classification accuracy only in a few datasets. These datasets generally have the same characteristics: They are both heavily imbalanced and small-sized. In such cases, there may not be enough samples in the dynamic selection dataset for the ENN filter and the AKNN to work properly. In fact, the ENN technique tends to remove instances from the minority class since they are under-represented, and some isolated instances may be considered as noise by the algorithm. Hence, we believe that the best strategy to deal with problems that are heavily imbalanced involves using a prototype generation technique, such as in [4, 24], to generate samples for the

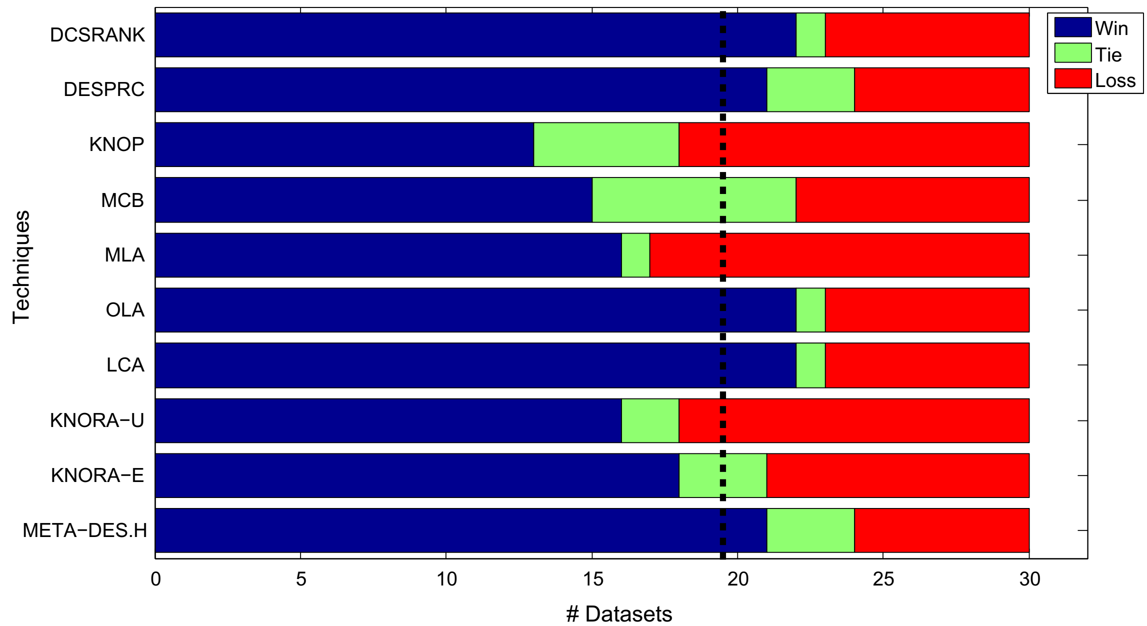

Fig. 5 Performance of the each dynamic selection technique using the ENN and A-KNN in terms of wins, ties and losses. The *dashed line* illustrates the critical value $n_c = 19.5$

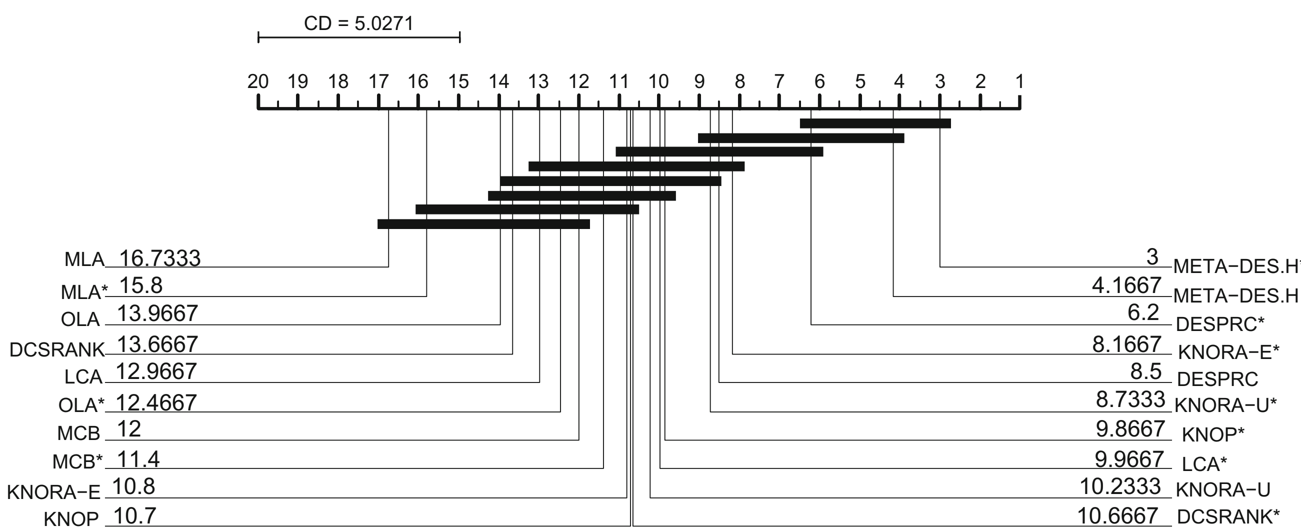

Fig. 6 CD diagram considering all techniques. Techniques marked with a *asterisk symbol* are the ones using Scenario IV

minority class, and apply the prototype selection only for the majority class.

Another important aspect of the proposed scheme is that, by removing samples in DSEL, the running time of the dynamic selection techniques decreases. For every technique, the running time to classify a given test instance $\mathbf{x}_j$ of each method is a combination of the definition of the region of competence and evaluating the competence level of each classifier in the pool. The definition of the region of competence is performed only once as it depends only on the input sample $\mathbf{x}_j$, and not on the base classifier. Since it is performed based on the AKNN technique, the cost is of order $O(d \times N)$, given that d and N are the number of dimensions and samples in the dynamic selection dataset (DSEL$'$), respectively.

For each dynamic selection technique, the outputs of the base classifiers for the samples in DSEL must first be pre-calculated during the training stage of the system and stored in a matrix. The storage requirement for the pre-calculated information is $O(M \times N \times \Omega)$, with M and Ω being the number of classifiers in the pool and the number of classes in the dataset. The computational cost involved

Table 4 Comparison of the results achieved by META-DES framework [9], considering Scenarios I and IV

Dataset	META-DES.H*	META-DES.H
Pima	**78.80 (3.23)**	77.93 (1.86)
Liver	**70.73 (4.10)**	69.69 (4.68)
Breast	97.02 (0.49)	**97.25 (0.47)**
Blood	**79.85 (1.12)**	78.25 (1.37)
Banana	**95.51 (1.46)**	94.51 (2.36)
Vehicle	83.24 (2.05)	**83.55 (2.10)**
Lithuanian	**94.75 (2.71)**	93.26(3.22)
Sonar	**82.06 (5.74)**	82.06(5.09)
Ionosphere	**89.31 (2.94)**	89.06(2.21)
Wine	**99.02 (1.83)**	98.53(1.48)
Haberman	75.83 (1.58)	**76.13 (2.06)**
CTG	**86.89 (1.41)**	86.08 (1.24)
Vertebral	**87.47 (7.05)**	84.90(2.95)
Faults	**69.41 (1.36)**	68.95 (1.04)
WDVG1	84.63 (0.75)	**84.77 (0.65)**
Ecoli	**80.66 (3.48)**	**80.66 (3.48)**
GLASS	**65.21 (3.53)**	**65.21 (3.53)**
ILPD	**70.02 (2.82)**	69.64 (2.47)
Adult	**87.74 (2.84)**	87.29 (1.80)
Weaning	**81.15 (3.33)**	79.98 (3.55)
Laryngeal1	**87.63 (4.19)**	87.21 (5.35)
Laryngeal3	**74.17 (2.89)**	73.54 (0.67)
Thyroid	96.99(6.13)	**97.38 (1.66)**
German	**75.87 (2.59)**	74.36 (1.28)
Heart	**85.62 (3.34)**	85.46 (2.70)
Segmentation	**96.52 (1.06)**	96.46 (0.79)
Phoneme	**82.68 (1.31)**	81.82 (0.69)
Monk2	**92.40 (2.58)**	83.45 (3.46)
Mammographic	80.24 (8.61)	**84.30 (2.27)**
Magic	**86.02 (2.20)**	85.65(2.27)

Best results are highlighted

during generalization consists in accessing the outputs of the base classifier stored in the matrix and applying the selection criteria for each base classifier in the pool. Thus, the cost of evaluating the competence of each classifier in the pool of classifiers is $O(M)$.

Therefore, besides improving the classification accuracy, the proposed scheme can also reduce the memory requirement and the running time of dynamic selection techniques during the generalization phase.

4 Conclusion

In this paper, we demonstrate that the performance of DES techniques is sensitive to the dynamic selection dataset distribution. A high degree of overlap in the dynamic selection dataset may lead to poor estimations of the local competence of the base classifiers; thus, the dynamic selection technique fails to select the most appropriate classifier for the classification of a new query sample. We show that with two simple modifications, we can significantly improve the generalization performance of any dynamic selection technique.

In order to evaluate the impact of the proposed scheme, we compared the results of ten dynamic classifier selection and dynamic ensemble selection techniques over 30 classification datasets. The experimental results demonstrate that the proposed scheme significantly improves the classification accuracy of the dynamic selection techniques. The scenario using both the ENN and A-KNN techniques presented the overall best result. In addition, using only the ENN for editing the dynamic selection dataset brings about a significant gain in classification accuracy.

Future work will include the evaluation of different prototype selection techniques, as well as prototype generation for dealing with problems that are both small sized and heavily imbalanced.

Acknowledgments This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), the École de technologie supérieure (ÉTS Montréal) and CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico).

References

1. Alcalá-Fdez J, Fernández A, Luengo J, Derrac J, García S (2011) KEEL data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. Mult Valued Log Soft Comput 17(2–3):255–287
2. Bache K, Lichman M (2013) UCI machine learning repository. http://archive.ics.uci.edu/ml
3. Britto AS, Sabourin R, de Oliveira LES (2014) Dynamic selection of classifiers: a comprehensive review. Pattern Recognit 47(11):3665–3680
4. Calvo-Zaragoza J, Valero-Mas JJ, Rico-Juan JR (2016) Prototype generation on structural data using dissimilarity space representation. Neural Comput Appl. doi:10.1007/s00521-016-2278-8
5. Cavalin PR, Sabourin R, Suen CY (2012) Logid: an adaptive framework combining local and global incremental learning for dynamic selection of ensembles of HMMs. Pattern Recognit 45(9):3544–3556
6. Cavalin PR, Sabourin R, Suen CY (2013) Dynamic selection approaches for multiple classifier systems. Neural Comput Appl 22(3–4):673–688
7. Cruz RMO, Sabourin R, Cavalcanti GDC (2015) A DEEP analysis of the META-DES framework for dynamic selection of ensemble of classifiers. CoRR arXiv:1509.00825
8. Cruz RMO, Sabourin R, Cavalcanti GDC (2015) META-DES.H: a dynamic ensemble selection technique using meta-learning and a dynamic weighting approach. In: International joint conference on neural networks, pp 1–8
9. Cruz RMO, Sabourin R, Cavalcanti GDC, Ren TI (2015) META-DES: a dynamic ensemble selection framework using meta-learning. Pattern Recognit 48(5):1925–1935

10. Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7:1–30
11. Díez-Pastor J, Rodríguez JJ, García-Osorio C, Kuncheva LI (2015) Diversity techniques improve the performance of the best imbalance learning ensembles. Inf Sci 325:98–117
12. Dos Santos EM, Sabourin R, Maupin P (2008) A dynamic overproduce-and-choose strategy for the selection of classifier ensembles. Pattern Recognit 41:2993–3009
13. Duin RPW, Juszczak P, de Ridder D, Paclik P, Pekalska E, Tax DM (2004) Prtools, a matlab toolbox for pattern recognition http://www.prtools.org
14. Garcia S, Derrac J, Cano J, Herrera F (2012) Prototype selection for nearest neighbor classification: taxonomy and empirical study. IEEE Trans Pattern Anal Mach Intell 34(3):417–435
15. Giacinto G, Roli F (2001) Dynamic classifier selection based on multiple classifier behaviour. Pattern Recognit 34:1879–1881
16. King RD, Feng C, Sutherland A (1995) Statlog: comparison of classification algorithms on large real-world problems. Appl Artif Intell Int J 9(3):289–333
17. Ko AHR, Sabourin R, Britto uS Jr (2008) From dynamic classifier selection to dynamic ensemble selection. Pattern Recognit 41:1735–1748
18. Kuncheva L (2004) Ludmila kuncheva collection LKC. http://pages.bangor.ac.uk/~mas00a/activities/real_data.htm
19. Nanni L, Fantozzi C, Lazzarini N (2015) Coupling different methods for overcoming the class imbalance problem. Neurocomputing 158:48–61
20. Sabourin M, Mitiche A, Thomas D, Nagy G (1993) Classifier combination for handprinted digit recognition. In: International conference on document analysis and recognition, pp 163–166
21. Smith MR, Martinez TR, Giraud-Carrier CG (2014) An instance level analysis of data complexity. Mach Learn 95(2):225–256
22. Smits PC (2002) Multiple classifier systems for supervised remote sensing image classification based on dynamic classifier selection. IEEE Trans Geosci Remote Sens 40(4):801–813
23. Sun Z, Song Q, Zhu X, Sun H, Xu B, Zhou Y (2015) A novel ensemble method for classifying imbalanced data. Pattern Recognit 48(5):1623–1637
24. Triguero I, Derrac J, García S, Herrera F (2012) A taxonomy and experimental study on prototype generation for nearest neighbor classification. IEEE Trans Syst Man Cybern Part C 42(1):86–100
25. Valentini G (2003) Ensemble methods based on bias-variance analysis. Ph.D. thesis, Dipartimento di Informatica e Scienze dell' Informazione (DISI)—Università di Genova
26. Wang J, Neskovic P, Cooper LN (2007) Improving nearest neighbor rule with a simple adaptive distance measure. Pattern Recognit Lett 28:207–213
27. Wilson DL (1972) Asymptotic properties of nearest neighbor rules using edited data. IEEE Trans Syst Man Cybern 2(3):408–421
28. Woloszynski T, Kurzynski M (2010) A measure of competence based on randomized reference classifier for dynamic ensemble selection. In: International conference on pattern recognition (ICPR), pp 4194–4197
29. Woloszynski T, Kurzynski M (2011) A probabilistic model of classifier competence for dynamic ensemble selection. Pattern Recognit 44:2656–2668
30. Woloszynski T, Kurzynski M, Podsiadlo P, Stachowiak GW (2012) A measure of competence based on random classification for dynamic ensemble selection. Inf Fusion 13(3):207–213
31. Woods K, Kegelmeyer WP Jr, Bowyer K (1997) Combination of multiple classifiers using local accuracy estimates. IEEE Trans Pattern Anal Mach Intell 19:405–410
32. Zhu X, Wu X, Yang Y (2004) Dynamic classifier selection for effective mining from noisy data streams. In: Proceedings of the 4th IEEE international conference on data mining, pp 305–312