Fast Classification in Industrial Big Data Environments

Helene Dörksen, Uwe Mönks, and Volker Lohweg, Senior Member, IEEE
inIT – Institute Industrial IT
Ostwestfalen-Lippe University of Applied Sciences
Liebigstr. 87, D-32657 Lemgo, Germany
Email: {helene.doerksen, uwe.moenks, volker.lohweg}@hs-owl.de

Abstract—Many modern industrial applications, e.g. those incorporating hundreds or thousands of electrical sensors and actuators, must be categorised into Big Data environments, in which it is essential to design suitable information processing models. Central data processing in such environments is impossible and must be carried out in a distributed way on resource-limited cyber-physical systems. One of the challenging tasks for machine learning is thus the design of a classifier which is simple, accurate and has an acceptable realisation time. We present ComRef-2D-ConvHull method for linear classification optimisation in lower-dimensional feature space, which is based on ComRef from [1]. Compared to original ComRef, we consider only classification optimisation in 2-dimensional feature spaces in ComRef-2D-ConvHull. Due to the decreased time complexity for calculations in 2-dimensional feature space, we expect many industrial Big Data environments to profit from our method. Tests regarding the generalisation ability of ComRef-2D-ConvHull on several reference data sets and on a real-world industrial dataset show promising results.

I. INTRODUCTION

Large-scale industrial machinery apply complex processes to cover the demands put on them. Such processes range from car manufacturing over newspaper printing up to garment weaving, to name just a few. The complexity of the machinery applied in such processes puts subsequent demands on their handling. In the engineering phase, modularisation approaches are followed to divide complex plants into smaller, less complex parts forming the desired process when operated in the appropriate combination. During operation the machine’s control, supervision, and service must be reliably guaranteed.

In this contribution, we concentrate on processes in which numerous actuators and sensors (many hundreds up to thousands) are applied. One example is industrial printing processes, like a newspaper printing process depicted in Fig. 1. Today’s state-of-the-art printing systems are driven by hundreds of actuators in the application, along with a number of sensors in the same order of magnitude. On the one hand, these are electric drives moving the conveyor belts, or magnetic valves controlling the print colour’s application onto the substrate, for example.

On the other hand, a vast variety of sensory units are applied for acquisition of different types of data. These may be several different basic physical measures such as pressure or temperature, but also specific process parameters like the quality of the fluid used to wipe off surplus print colour, as well as many others.

Even more actuators in form of electrical drives are applied in modern weaving machines. Here, typically around two thousand electric motors drive the process and move the yarnwinders, needles, and woven garments.

For a guaranteed reliable operation of each process, all of its parts must be monitored continuously. Here, novel sensory concepts which turn actuators into sensors are introduced nowadays. One example is the Motor as Sensor (MaS) concept [2]: The trend in the field of electric drives is towards integrated systems, in which the power and control electronics are part of the electric motor itself. In these systems, sensors are built-in to control the drive. The conceptual idea of the MaS concept is to supply the information obtained by those sensors to systems outside the electric drive.

Within the two exemplary scenarios mentioned above, the challenges arising when it comes to machine condition monitoring can be clearly pointed out. Besides sensors also hundreds or thousands of actuators are inside these systems. Therefore, the amount of data generated and acquired literally explodes by the addition of X-as-Sensor approaches. The data consequently is subject to the 3Vs following Laney’s Big Data definition [3]:

**Variety** In complex industrial production processes, various kinds of physical parameters are captured by sensors, enriched by the signals supplied by the manifold actuators. This ends up in hundreds of different data types occurring in such applications.

**Velocity** Process control units as well as machine condition monitoring systems acquire data at sample rates in the range of some kSample/s. This is absolutely necessary to facilitate (i) stable process control fulfilling typical real-time constraints and (ii) reliable machine operation preventing production faults by means of condition monitoring systems.

**Volume** Assuming a typical data source’s (sensor or actuator) resolution of 10 bit and sample rate of $16 \cdot 10^3$ Sample/s, $160 \cdot 10^3$ bit/s of raw data are provided by each source applied in the production system. With 100 sources, the data rate occurring per production system is at $16 \cdot 10^6$ bit/s, hence around 1.91 MB/s or 160.93 GB/day raw data is generated.

Thus, industrial production processes must clearly be categorised as Big Data applications. Manual human condition monitoring processing all the data generated is therefore impossible and must be carried out by proper multisensor
information fusion systems. Since the signal sources are distributed over the complete application, all the data must usually be communicated over an appropriate network to the condition monitoring system. The above considerations regarding volume have been carried out quite conservatively. Nevertheless, already these conservative figures point out that centralised condition monitoring systems are not able to handle all occurring data due to restrictions of current fieldbus systems needed to communicate the data. The situation additionally deteriorates when instead of one single process a complete plant consisting of a number of production machines is to be monitored. Such systems are in the end not scalable.

Our proposed solution to counter these challenges is the introduction of local intelligence into sensor and actuator units for data reduction. The implanted intelligence facilitates self-X properties [4] of the machine’s devices. As such, the devices are carried out as cyber-physical systems (CPS) [5] to process the generated data, diagnose the device’s state (e.g., wearout degree), and adapt to fluctuating process or environment conditions on their own. Consequently, only regular or event-based state data instead of raw data needs to be transmitted to a central monitoring system, resulting in a drastic reduction of the bandwidth demands towards communication.

One major challenge in the realisation of CPS for production systems (of which an ensemble forms cyber-physical production systems [7]) is the resource limitation of such embedded systems. Current integrated electric drives for example already carry processing units for their own control, but due to cost constraints these processing units with accompanied memory have little or no resources left for additional tasks to be carried out on the devices. Consequently, only small extensions of the processing units may be allowed for integration of monitoring applications.

Under these constraints, established signal processing algorithms and systems are at their frontiers, if not even inapplicable in such scenarios prone to resource limitations together with appearance of big data amounts. The contribution we are presenting here is an execution-time optimised robust classifier, being a crucial part in a signal processing chain as a classifier has to deliver trustful results in an industrial environment.

The paper is structured as follows: Related work is summarised and discussed in the following Sect. II. In Sect. III we present our optimised classification approach, which is evaluated in Sect. IV. The paper is concluded in Sect. V.

II. RELATED WORK

As discussed above, a worldwide trend to increasingly complex systems for process automation, strongly driven by computer science and information technology (IT), can be observed, which legitimates the handling of unstructured information in the context of “Big Data”. Two main threads are located in the context of those approaches: (i) Towards the Internet of Data and Services from central computer over data warehousing to cloud computing and smart devices and (ii) towards the Internet of Things via physical objects and embedded systems to cyber-physical systems [7]. These two main threads will converge in the next years.

In the following we give some insights regarding tools for handling “big” data volumes: MapReduce [8] is a programming model for parallel computation of large data volumes on computer clusters which was launched by Google Inc. Apache Hadoop [9] is used as a scalable, distributed software based on MapReduce in a Java framework. The MapReduce applicability for machine learning (ML) was described by [10], [11] in 2006. Furthermore, it was shown that the interconnection of Hadoop systems with the statistical analysis software R is possible [12]. Another open-source framework for distributed data based on MapReduce is DISCO (Distributed Collaborative Aggregation) [13]. DISCO utilises scalable collaborative aggregation techniques and is able to handle some 100 GB of data parallel. However, the above mentioned systems need computer clusters for the processing of unstructured high volume data, whereas our approach assumes that data is processed at the point-of-occurrence (inside a machine).

Subsequently the data is fused for systems state determination purposes, but sensor and information fusion is established in the industrial area [14] to a small extent. However, due to the requirements towards robustness and reliability no application-specific fusion approaches for resource-limited embedded systems are known up to now. This is also because no generalised, easy to implement and maintenance-free fusion approach has been established. Nevertheless, known fusion concepts introduce new possibilities to enable process quality stabilisation, robust decisions regarding system states, and therefore anticipating behaviour prediction [15]. Transferred to multiple electric drives, which means information from different drives must be shared among each other and fused subsequently. Applied fusion methods, e.g., parametrical approaches, consist of feature-based, probabilistic, fuzzy-based, and neural concepts. A variety of possible methods are discussed in [16]. One such fusion method is Dempster’s and Shafer’s evidence theory [17], which is capable to incorporate epistemic uncertainties by extending the classical probability theory and introducing upper and lower probabilities. These allow the distinction between the belief in an event and its plausibility. Thus, a number of heterogeneous information sources (e.g. sensors, experts, databases) are combined under consideration of each source’s credibility, and inherent conflict between the sources is reduced at the same time [18].
Handling the machinery’s complexity comes also into effect during the plant engineering process. As of today, production floors are engineered in a time-consuming tedious manual task by an expert. This holds for the machine engineering part as well as for the communication setup and the design of the machine condition monitoring system. Furthermore, each and every change of a production system must be integrated manually: plug and play mechanisms known from the office IT world are yet not established in the industrial area [19]. The German leading-edge cluster “it’s OWL – Intelligent Technical Systems Ostwestfalen-Lippe” funded by the Federal Ministry of Education and Research (BMBF) addresses these aspects in the research project “Intelligent Networking” [20]. Here, self-configuration mechanisms are defined and evaluated such that the communication inside and between production systems is set up in an automated way, especially in cases where the system is extended or some production module is removed. The information is also used for the (re-)configuration of the machine condition monitoring system applied for self-diagnosis, hence determining the production system’s state [21].

Additionally, great importance is attended on defining a sensor and information fusion system for self-diagnosis which resembles the production facility’s physical structure, resolves (or at least reduces) conflicts between information sources, and is implementable in resource-limited cyber-physical systems. The monitoring approach called multi-layer attribute-based conflict-reducing observation system MACRO, cf. Fig. 2, determines the state of a complete system from acquired sensor signals of the system itself as well as its environment [21]. Features are extracted from the sensors’ signals in the following signal conditioning step which may also include signal pre-processing procedures. Ensembles of conditioned signals are then grouped to so-called attributes representing certain properties or physical parts of the observed system or process. Redundancies occurring by combining two (or even more) information sources to one attribute are used beneficially for both (i) intercepting sensor faults and (ii) cross-checking the consistency of sensor values. The latter is carried out implicitly by μBaTLC fusion [22] on attribute layer: it creates one output signal per attribute from n input signals and connects the attribute with an importance measure, which is the negated conflict between the sensors’ individual opinions. Subsequently, the fused attributes’ opinions (μBalTLCS outputs) are aggregated on system level using the Implicative Importance Weighted Ordered Weighted Averaging (IIWOWA) operator [23]. This step is used to reason about the complete system under supervision, summarising all individual attributes. Details regarding MACRO’s definition and properties are found in [18], [22]. It recently made a big step towards CPS implementability by improving the computational complexity of μBaTLC fusion (one of MACRO’s core parts) from $O(n^2)$ to $O(n)$ [24].

As information fusion must be carried out application-specific in any case, the fusion approach used in the MACRO system on attribute layer might not fit best in some applications. In the following section we describe a possible substitute based on support vector machine approach. Support vector machines (SVM) received enormous attention in the past decade since so-called kernel functions were re-discovered [25] and the kernel trick was applied for non-linear classification. Recent developments on SVM applications are found in [26], for example. SVMs can produce models with different kinds of decision boundaries—linear up to highly non-linear. Due to the margin optimisation, the complexity of the border does not necessarily announce poor generalisation [27]. However, as it was discussed in [1], simple classification models are preferred, i.e. linear over non-linear, or those with the smallest number of parameters, etc. For Big Data scenarios, it means that models being simple, accurate, and with acceptable time consumption are required. In the next section, we present an approach, which is expected to be suitable for several Big Data sample sets.

III. Approach

In this section we present the application of the Combinatorial Refinement (ComRef) approach [1] as an efficient solution of the problems discussed above. ComRef performs the refinement of the chosen classifier in lower-dimensional spaces such that the number of correctly classified objects increases. From the simplicity point of view, ComRef has the same properties as the chosen classifier. Moreover, it was shown in [1] by $5 \times 2$ cv F cross-validation test [28], that in many cases ComRef is able to improve the generalisation ability of the classifier.

In the framework of this paper we discuss the application of ComRef with linear SVM as the main classifier. Without loss of generality, SVM separating hyperplane will be calculated using Sequential Minimal Optimization [29]. We consider the classification problem of two classes $x^+ \in T^+$ and $x^- \in T^-$, where $x^+, x^-$ are sets of standardised ($\sigma = 1$) vectors $(x_1, \ldots, x_d)$ in the feature space $X \subseteq \mathbb{R}^d$ [27]. We assume that the linear combination $h$ of features is calculated within the linear SVM method, i.e. for parameters $a_i$, $i = 1, \ldots, d$ we have:

$$h(x) = \langle a, x \rangle = \sum_{i=1}^{d} a_i x_i,$$  

(1)

where $\langle \cdot, \cdot \rangle$ denotes the inner product in the feature space. With the scalar $c \in \mathbb{R}$, the linear classification rule w.r.t. Eq. (1) is the following:

$$x \in T^+ \text{ if } h(x) \geq c \quad \text{and} \quad x \in T^- \text{ if } h(x) < c.$$  

(2)
For ComRef in [1] we defined fusions of summands of \( h \) for some indices \( I \subseteq \{1, \ldots, d\} \) as

\[
u_I := \sum_{i \in I} a_i x_i.
\]

For \( I = \{1, \ldots, d\} \), the fusion of summands is \( h \) itself. Otherwise, \( h \) can be represented by \( I \) together with the fusion of summands for the indices \( \bar{I} = \{1, \ldots, d\} \setminus I \), which are complementary to \( I \). More general, \( h \) can be represented by fusion of summands (for ease of use read \( u_I = u_k, k \in \mathbb{N} \)):

\[
h(x) = \sum_{i \in I_1} a_i x_i + \cdots + \sum_{i \in I_j} a_i x_i = u_1 + \cdots + u_j,
\]

where \( I_k \subseteq \{1, \ldots, d\} \forall k = 1, \ldots, j \) and all \( I_k \) are non-empty disjoint subsets of indices with

\[
\bigcup_{k=1}^j I_k = \{1, \ldots, d\}.
\]

In the ComRef approach [1] we are searching for fusions of summands based on Eq. (1) in lower-dimensional space \( \mathbb{R}^j \), \( j < d \), such that for the function

\[
g(u) = \langle b, u \rangle, \quad u \in \mathbb{R}^j,
\]

the classification rule (performing better than Eq. (2)) is found. The vector \( b \in \mathbb{R}^j \) is chosen to optimise the classification result. In that case, with some scalar \( \tilde{c} \in \mathbb{R} \) the new classification rule is:

\[
u \in T^+ \text{ if } g(u) \geq \tilde{c} \quad \text{and} \quad u \in T^- \text{ if } g(u) < \tilde{c}.
\]

Our aim is to use ComRef and find a more trustful classification rule (resp. Eq. (5)) within acceptable computational time. In this framework, we use SVM in Eq. (1), so it is appropriate to use SVM in Eq. (4) as well. From the time complexity point of view, it is a well known fact, that SVM can be calculated within \( O(m^3) \), where \( m = \sum I \) is the number of objects in the sample set.

**Proposition 1.** Let \( d \) be the dimension of the feature space and \( m \) be the number of objects in the dataset. The largest and smallest dimension of a reduced feature space is defined as: \( d - 1 \), resp. 2. For the ComRef approach the search for the best fusion of summands for Eq. (4) with respect to SVM classification has time complexity

(i) \( O(m^3d^2) \) for \( (d - 1) \)-dimensional space,

(ii) \( O(m^32^d) \) for 2-dimensional space,

within the dimension borders of the reduced feature space.

**Proof:** The statement follows directly from the fact that time complexity of SVM is \( O(m^3) \) and that there are \( d^2 \) fusions of summands in \( (d - 1) \)-dimensional space, resp. \( 2^d \) fusions of summands in 2-dimensional space [1].

In our Big Data scenario, the number of features \( d \) is considerably lower than the number \( m \). Thus, the contribution of \( m \) to the time complexity is stronger than of \( d \) (e.g., for \( d = 10 \) and \( m = 1000 \): \( 2^{d} = 1024 < 10^{4} \), but \( 1000^{3} = 10^{9} \)).

Regarding the number of objects \( m \) of the dataset, we are interested in the possibility to operate ComRef in a more efficient way. Before we announce the main statement of this paper, we introduce the definition of the convex set and convex hull of a set:

**Definition 1.** A set \( S \) in a feature space \( \mathbb{R}^d \) is called convex if for any \( x, y \in S \) and any \( \lambda \in [0, 1] \), we have

\[
\lambda x + (1 - \lambda)y \in S.
\]

The minimal convex set containing \( S \) is called convex hull of \( S \).

It is a well known fact from computational geometry that the convex hull computation of a finite set of points in \( \mathbb{R}^2 \) takes \( O(m \log m) \) time [30]. Another useful fact for our investigations is the relation between SVM and the vertices of the convex hull called extreme points, i.e., for the separable case finding the maximum margin between the two sets is equivalent to finding the closest points in the smallest convex sets that contain each class (the convex hulls) [31]. The relation between SVM and extreme points of the convex hull (more precisely, reduced convex hull) in the non-separable case is discussed in [31] as well. In our framework, we concentrate only on the separable case, since it is suitable for our application. The non-separable case will be a topic of our further research. Now we state:

**Lemma 1.** Let \( d = 2 \) be the dimension of the feature space and \( m \) be the number of objects in the dataset. Assume there exists a fusion of summands such that the classes are separable in \( \mathbb{R}^2 \) with respect to the fusion. For the ComRef approach the search for the mentioned fusion of summands has the time complexity \( O(\tilde{m}^32^d) \) with \( \tilde{m} \leq m, \tilde{m} \in \mathbb{N}^+ \).

**Proof:** From Prop. 1 we know that the time complexity for the calculation of fusions of summands in 2-dimensional space is \( O(m^32^d) \). The convex hull computation of the sample set for a fusion of summands in 2-dimensional space is executed within \( O(\tilde{m} \log \tilde{m}) \), where \( \tilde{m} \) is the number of extreme points. The procedures (convex hull calculation and SVM for extreme points) need \( O(\tilde{m} \log \tilde{m}) + O(\tilde{m}^3) \Rightarrow O(\tilde{m} \log \tilde{m} + \tilde{m}^3) \Rightarrow O(\tilde{m}^3) \). It is clear that, calculated on the extreme points, SVM’s classification rate for the separable case is higher than the rate for the non-separable case, i.e., we can restrict ComRef search to only the search on the extreme points in 2-dimensional spaces. As \( \tilde{m} \leq m \) holds for the number of extreme points, the proof of the lemma is fulfilled.

**Example.** We show the principle of ComRef with respect to SVM and convex hull computation in 2-dimensional spaces on the simple example Seeds from UCI repository [32]. The number of features is 7 and there are 70 objects in each class. The parameters \( a_i \) and \( c \) in Eq. (1) w.r.t. Eq. (2) are calculated using SVM. For the sample set we have the accuracy rate 95.71%. We consider ComRef for 2-dimensional spaces. In general, for different fusions of summands, we have different appearances of the sample set in the spaces. Furthermore, the sample set has different discriminative properties. Figures 3 and 4 illustrate this context. SVM for the sample set, w.r.t. Eqs. (4) and (5) for complete set and for extreme points of the convex hull, is indicated as well.

We summarise the procedure of our ComRef-2D-ConvHull-approach, which is based on [1]. The approach
takes into account only fusions of summands in 2-dimensional spaces and uses extreme points of convex hulls for parameter calculations in Eq. (4) and Eq. (5):

1) Initialise a linear combination $h$ of features for Eq. (1) with parameters $a_i, i = 1, \ldots, d$; assign $c$ for Eq. (2).

2) Calculate non-equal fusions of summands for 2-dimensional spaces as follows:

- if $d$ is even:
  - for all $j = 1, \ldots, \frac{d}{2} - 1$ and for all $p \in \wp(d, j)$ (where $\wp(d, j)$ denotes the set of all subsets of $\{1, \ldots, d\}$ of length $j$) assign $I_1 = p$, $I_2 = \{1, \ldots, d\} \setminus p$;
  - start with all subsets $P := \wp(d, \frac{d}{2})$ and for some $p \in P$ set $I_1 = p$, $I_2 = \{1, \ldots, d\} \setminus p$, update $P := P \setminus \{I_1, I_2\}$ and repeat setting and updating until $P = \emptyset$;
- if $d$ is odd: for all $j = 1, \ldots, \frac{d-1}{2}$ and for $p \in \wp(d, j)$ set $I_1 = p$, $I_2 = \{1, \ldots, d\} \setminus p$.

3) For fusions of summands calculate features $u_1, u_2$.

4) Regarding features $u_1, u_2$, calculate convex hull of the sample set.

5) Choose a classification approach; use extreme points of the convex hull for calculating parameters $b_1, b_2$ and $\hat{c}$ for Eq. (4) and Eq. (5).

6) With respect to complete sample set, select one fusion of summands with best classification rate or best margin.

7) For selected fusion of summands calculate $g$, which is:

\[ g = b_1 \left( \sum_{i \in I_1} a_i x_i \right) + b_2 \left( \sum_{i \in I_2} a_i x_i \right) \]

and set ComRef-2D-ConvHull-approach classification rule:

\[ x \in T^+ \text{ if } g(x) \geq \hat{c} \quad \text{ and } \quad x \in T^- \text{ if } g(x) < \hat{c}. \]

IV. EXPERIMENTS AND RESULTS

In this section we test the performance of ComRef-2D-ConvHull on several UCI samples and a real-world example from an industrial application. Compared to the UCI sample sets, the industrial example is of much larger sample size. It is not the intention of our paper to optimise classification rates as such, but to speed up the classifier training. Therefore, the classification rates are partly improved, though they stay mainly in the range of the results which are generated by SVM. All experiments were carried out using Matlab R2012a on 64bit Windows® PC equipped with an Intel® Core™ i7-3820 CPU@3.60 GHz and 16.0 GB RAM.

A. Generalisation Abilities and Time Consumption for Test Samples

We have tested generalisation abilities and time consumption of ComRef-2D-ConvHull on several datasets. The generalisation performance was explored by $5 \times 2$ cv $F$ cross-validation method (meaning five times 2-fold cross-validation [33]). The original of the method is cited in [28]. It was pointed out in [28] that after five folds, the sets overlap and share many statistics implying that new folds do not add new information.

The following datasets from UCI repository [32] are used: Seeds (class 1 vs. class 2, i.e. 140 objects are considered), Iris (versicolor vs. virginica, i.e. 100 objects are considered), Mammography, Liver, Indian liver, Breast cancer, Tic-Tac-Toe and Vertebral. The presented datasets with maximal 961 objects are used to show that (in many cases) ComRef-2D-ConvHull is able to outperform SVM or ComRef. Furthermore, the results of the dataset Motor Drive Diagnosis below approves that ComRef-2D-ConvHull might be suitable to the datasets with a much larger number of measurements.

Table I represents classification rates of datasets from UCI repository for $5 \times 2$ cv $F$ test in terms of accuracies (in %) and their standard deviations. Time consumptions (in sec.) of original ComRef and ComRef-2D-ConvHull are given. We have to remark here that results of Table I for accuracies and standard deviations might appear slightly different from what is presented in [1]. The differences originate from random splits of the sample into training and test subsets for the cross-validation.

From the results we deduce that, regarding accuracies and standard deviations, ComRef-2D-ConvHull outperforms SVM in many cases. Moreover, for some cases ComRef-2D-ConvHull outperforms accuracies and standard deviations of ComRef. For the considered datasets, the time savings
of ComRef-2D-ConvHull compared to ComRef vary between 21.69 % and 84.51 %.

### B. Motor Drive Diagnosis

We illustrate the advantage of the method ComRef-2D-ConvHull using data from the industrial real-world application referred to as Motor Drive Diagnosis [34]. To be comparable to the original ComRef, the same example was analysed in [1].

We recall the problem description from [1]. For the **complete sample set** consisting of 53190 measurements with 72 features each, classification accuracy for the linear SVM is 100 %. Two classes are considered—**intact** and **anomaly**—for the motor condition, which is a conclusive result, taking into account the use of adequate features [35]. Moreover, the examinations have shown that there are **several feature subsets** < 72 features which perform with 100 % accuracy as well (on the **complete sample set**). Since for the application already a small number of mis-classifications could lead to total motor damage and negative financial consequences, a trustful classifier with high generalisation rates is needed, i.e., the most trustful feature subset has to be taken. The experiments have shown that using the entire 72 features a classifier generalises worse than on some of the subsets. Generalisation abilities of subsets and use of ComRef-approach for the selection of the most trustful ones was shown in [1]. From the results presented in Table II we deduce, that using ComRef-2D-ConvHull, we are able to find the same most trustful subset as provided by ComRef. However, the time consumption for ComRef-2D-ConvHull is reduced by 78 %. Regarding the time reduction, we are expecting ComRef-2D-ConvHull technique to be attractive for many industrial Big Data environments.

We remark, that the **Motor Drive Diagnosis** data set presented here will be submitted to the UCI repository soon.

### V. Conclusion and Outlook

Regarding machine learning tasks for industrial Big Data applications, information processing models are required which should be as simple as possible, accurate and with acceptable time consumption to realise.

In this paper we presented an approach called ComRef-2D-ConvHull for linear classification optimisation in lower-dimensional feature spaces. ComRef-2D-ConvHull is a special case of the ComRef approach from [1]. In difference to ComRef, ComRef-2D-ConvHull considers only classification optimisation for fusions of summands (as defined in Eq. (3)) leading to 2-dimensional feature spaces. In 2-dimensional feature space, from the time consumption point of view, for some datasets (i.e., in the case the classes are separable) the classification optimisation can be efficiently performed on the extreme points of the convex hull. Due to the decreased time complexity for calculations in 2-dimensional feature space, we are expecting that many industrial Big Data applications will profit from our method.

Experimental results based on UCI data sets and the real-world **Motor Drive Diagnosis** data set have shown that in many cases ComRef-2D-ConvHull is able to improve the generalisation ability of a classifier or/and of ComRef, meaning that more trustful results can be achieved. Furthermore, comparing to original ComRef approach, ComRef-2D-ConvHull is able to reduce the time consumption significantly for tested data sets. From the theoretical point of view, ComRef-2D-ConvHull is primarily suitable for data sets containing separable classes. Modifications of ComRef-2D-ConvHull for the non-separable case will be the aim of our further research. Furthermore, statistical methods for fast searching for the best fusion of summands will be examined. The fast searching for the best fusion of summands will take the refinement approach to

<table>
<thead>
<tr>
<th>dataset</th>
<th>#feat.</th>
<th>#obj.</th>
<th>SVM</th>
<th>ComRef</th>
<th>time [s]</th>
<th>ComRef-2D-ConvHull</th>
<th>time [s]</th>
<th>speed-up factor [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seeds</td>
<td>7</td>
<td>140</td>
<td>97.34 (1.61)</td>
<td>97.47 (1.65)</td>
<td>5.36</td>
<td>97.47 (1.59)</td>
<td>1.72</td>
<td>67.91</td>
</tr>
<tr>
<td>Iris</td>
<td>4</td>
<td>100</td>
<td>93.33 (3.34)</td>
<td>94.17 (2.33)</td>
<td>0.83</td>
<td>93.83 (2.62)</td>
<td>0.65</td>
<td>21.69</td>
</tr>
<tr>
<td>Breast cancer</td>
<td>9</td>
<td>683</td>
<td>96.53 (0.99)</td>
<td>96.53 (0.95)</td>
<td>88.38</td>
<td>96.57 (0.99)</td>
<td>14.51</td>
<td>83.58</td>
</tr>
<tr>
<td>Tic-Tac-Toe</td>
<td>9</td>
<td>958</td>
<td>58.06 (2.20)</td>
<td>64.57 (3.78)</td>
<td>144.55</td>
<td>58.96 (1.36)</td>
<td>22.39</td>
<td>84.51</td>
</tr>
<tr>
<td>Liver</td>
<td>6</td>
<td>345</td>
<td>64.59 (4.74)</td>
<td>64.72 (4.51)</td>
<td>27.39</td>
<td>64.80 (4.46)</td>
<td>5.82</td>
<td>78.75</td>
</tr>
<tr>
<td>Mammography</td>
<td>5</td>
<td>961</td>
<td>81.46 (1.89)</td>
<td>81.55 (1.96)</td>
<td>15.85</td>
<td>80.75 (1.64)</td>
<td>3.84</td>
<td>75.77</td>
</tr>
<tr>
<td>Indian liver</td>
<td>10</td>
<td>579</td>
<td>61.10 (1.63)</td>
<td>62.14 (1.75)</td>
<td>304.94</td>
<td>61.73 (1.54)</td>
<td>52.18</td>
<td>82.89</td>
</tr>
<tr>
<td>Vertebral</td>
<td>6</td>
<td>310</td>
<td>82.50 (2.04)</td>
<td>82.50 (2.20)</td>
<td>14.87</td>
<td>82.54 (2.01)</td>
<td>3.88</td>
<td>73.91</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>subset</th>
<th>#feat.</th>
<th>SVM</th>
<th>ComRef</th>
<th>time [s]</th>
<th>ComRef-2D-ConvHull</th>
<th>time [s]</th>
<th>speed-up factor [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>9</td>
<td>99.9933 (0.0095)</td>
<td>99.9989 (0.0049)</td>
<td>320.85</td>
<td>99.9989 (0.0049)</td>
<td>40.90</td>
<td>87.25</td>
</tr>
<tr>
<td>II</td>
<td>6</td>
<td>99.9761 (0.0202)</td>
<td>99.9863 (0.0173)</td>
<td>73.02</td>
<td>99.9863 (0.0173)</td>
<td>13.40</td>
<td>81.65</td>
</tr>
<tr>
<td>III</td>
<td>4</td>
<td>99.9861 (0.0156)</td>
<td>99.9941 (0.0088)</td>
<td>23.49</td>
<td>99.9941 (0.0088)</td>
<td>7.50</td>
<td>68.07</td>
</tr>
<tr>
<td>IV</td>
<td>5</td>
<td>99.9858 (0.0186)</td>
<td>99.9840 (0.0179)</td>
<td>40.40</td>
<td>99.9737 (0.0283)</td>
<td>7.41</td>
<td>81.66</td>
</tr>
<tr>
<td>V</td>
<td>5</td>
<td>99.9866 (0.0187)</td>
<td>100.00 (0.00)</td>
<td>33.87</td>
<td>100.00 (0.00)</td>
<td>7.42</td>
<td>78.09</td>
</tr>
</tbody>
</table>
be applicable to the datasets with a much larger number of features as presented in the framework.

ACKNOWLEDGMENT

We would like to thank Martyna Bator of Institute Industrial IT for providing helpful support regarding Motor Drive Diagnosis dataset. This work was partly funded by the German Federal Ministry of Education and Research (BMBF) within the Leading-Edge Cluster “Intelligent Technical Systems Ost-WestfalenLippe” (it’s OWL).

REFERENCES


