Ensemble Learning for Multi-source Information Fusion

Jörg Beyer[1,2], Kai Heesche[1], Werner Hauptmann[1], Clemens Otte[1], and Rudolf Kruse[2]

[1] Siemens AG - Corporate Technology, Information and Communications, Learning Systems, Otto-Hahn-Ring 6, 80200 Munich, Germany
[2] Otto-von-Guericke-University Magdeburg - School of Computer Science, Universitätsplatz 2, 39106 Magdeburg, Germany

Abstract. In this paper, a new ensemble learning method is proposed. The main objective of this approach is to jointly use knowledge-based and data-driven submodels in the modeling process. The integration of knowledge-based submodels is of particular interest, since they are able to provide information not contained in the data. On the other hand, data-driven models can complement the knowledge-based models with respect to input space coverage. For the task of appropriately integrating the different models, a method for partitioning the input space for the given models is introduced. The benefits of this approach are demonstrated for a real-world application.

1 Introduction

Real-world applications are characterized by an increasing complexity. To generate adequate models the consideration of all available information sources is necessary. For this purpose, more and more sophisticated combinations of knowledge-based and data-driven models are required which are representing these sources. While data-driven models are learned from available training data the integration of knowledge-based models is of particular interest since they are able to provide information not contained in the training data. The knowledge-based models are designed for particular regions of the input space. In order to ensure that the models are only active in regions they are designed for, their specific validity ranges have to be included in the modeling process.

The use of multiple submodels is motivated by the paradigm that different submodels can complement each other avoiding the weakness of a single model. The combination of models constitutes an ensemble as depicted in Fig. 1. According to the divide-and-conquer principle a complex task is solved by dividing it into a number of simpler tasks and then combining the solutions of those tasks. The ensemble fuses information y_j acquired by model j, $j = 1, \ldots M$, to produce an overall solution y that is supposedly superior to that attainable by any one of them acting alone. Literature describes many approaches that address the problem of learning local models. Examples of such methods are boosting [1], mixture of experts [2], or ensemble averaging [3]. The algorithms for learning

C. Sossai and G. Chemello (Eds.): ECSQARU 2009, LNAI 5590, pp. 748–756, 2009.

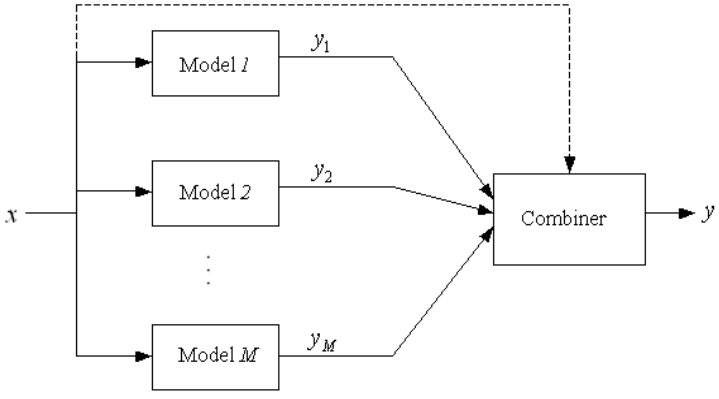

Fig. 1. A common ensemble model. The dashed line indicates that the Combiner can involve the current input in its decision dependent on the combining method.

local models can be discriminated with respect to several aspects: in the way they divide the training data into subsets, the type of submodels they use, or how they combine the outputs of the submodels. However, none of the existing methods are able to integrate predefined models that are designed for particular regions of the input space.

The paper is organized as follows: In Sect. 2, an introduction of multi-source fusion is given and Sect. 3 describes an ensemble learning model for combining data-driven and knowledge-based models. In Sect. 4, some experiments on a real-world application are outlined. Sect. 5 concludes the paper.

2 Multi-source Information Fusion

The term information fusion (IF) encompasses the process of merging and integrating heterogeneous information components from multiple sources, for instance, in the form of sensors, human experts, symbolic knowledge, or physical process models (according to Dasarathy [4]). IF is an important technique in different application domains, such as sensor fusion [5], identity verification [6], or signal and image processing [7].

Fusion implies the combination of information from more than one source. There are different reasons for fusion of multiple sources:

- The combined solution is able to attain more accurate, transparent, and robust results since the different information sources can complement each other with respect to their strengths and weaknesses.
- A model that depends on a single source is not robust with respect to error-proneness, i.e. if the single source is erroneous the whole model is affected. Models based on fused information sources are more robust since other sources are able to compensate for incorrect information.

– Fusion of information sources will provide extended coverage of information of the process to be modeled.

We consider two kinds of fusion approaches: complementary and cooperative fusion. They are discriminated with respect to the relationship among the information sources. In complementary fusion each source provides information from a different region of the input space, i.e. their responsibilities do not overlap. These sources provide locally a high performance. However, outside their regions the results are not valid. Cooperative fusion means that the information is shared among several information sources in the same region of the input space and has to be fused for a more complete modeling of the underlying process.

The next section describes an ensemble learning approach for IF. The information sources will be represented by predefined models. The process of partitioning the input space and the fusion of the models is performed by a separate data-driven model.

3 Combining Knowledge-Based and Data-Driven Models

The proposed ensemble model, referred to as heterogeneous mixture of experts (HME) model, is based on the mixture of experts (ME) approach [2], [8]. This model consists of a set of submodels that perform a local function approximation. The decomposition of the problem is learned by a gate function which partitions the input space and assigns submodels to these regions. In contrast to the ME model, the proposed ensemble learning method starts with some knowledge-based submodels, representing different information sources. Fig. 2 illustrates a general HME model. It consists of different models and a gate. To ensure that these submodels are assigned to those domains of the input space they are designed for, information about the specific validity ranges of the predefined knowledge-based submodels is used for the partitioning of the input space. It is assumed that the knowledge-based models will only cover a part of the input space while data-driven models learn the remainder.

From the probabilistic perspective the output of the HME model can be interpreted as the probability of generating output $y^{(n)}$ given input vector $\boldsymbol{x}^{(n)}$:

$$P\left(y^{(n)} \left| \boldsymbol{x}^{(n)}, \Theta \right.\right) = \sum_{j=1}^{M} P\left(z_j^{(n)} \left| \boldsymbol{x}^{(n)}, \theta_g \right.\right) P\left(y^{(n)} \left| \boldsymbol{x}^{(n)}, \theta_j \right.\right) \ , \tag{1}$$

where M is the number of submodels, Θ is the set of parameters $\left\{\theta_g, \{\theta_j\}_{j=1}^{M}\right\}$ of the gate and of the submodels, respectively. The input vector $\boldsymbol{x}^{(n)} \in \Re^k$ and the output $y^{(n)} \in \Re$, where $n = 1, \ldots N$. The probability $P\left(z_j^{(n)} \left| \boldsymbol{x}^{(n)}, \theta_g \right.\right)$ represents the mixture coefficient of model j. The latent variable $z_j^{(n)}$ indicates which input vector $\boldsymbol{x}^{(n)}$ was generated by model j. Its introduction simplifies the training algorithm and allows the HME to be trained with the Expectation-Maximization (EM) algorithm [9]. The probability $P\left(y^{(n)} \left| \boldsymbol{x}^{(n)}, \theta_j \right.\right)$ represents the conditional densities of target $y^{(n)}$ for model j.

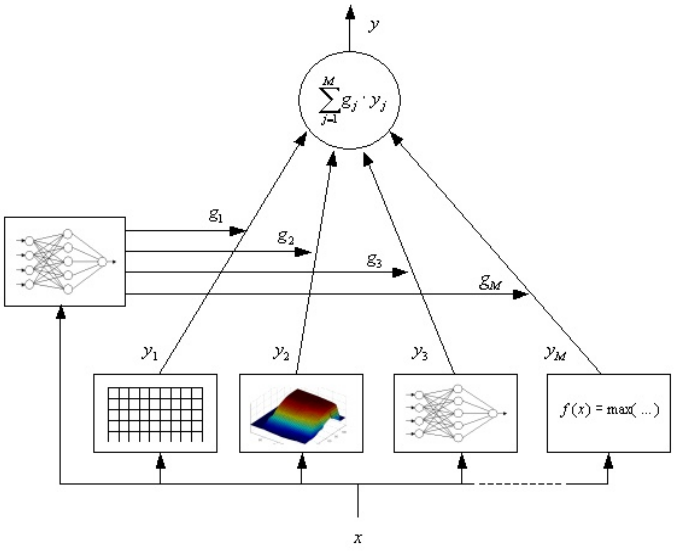

Fig. 2. Architecture of the proposed ensemble model. We include the case that gate and submodels may depend on different feature subsets of the input vector.

To compute the validity of each knowledge-based submodel j for an input vector a mapping $v_j : \Re^k \rightarrow [0, 1], \forall j = 1, \ldots, M$ is defined. The specific validity function of a knowledge-based submodel j for the i-th dimension is

$$v_j\left(x_i^{(n)}\right) = \left(\frac{1}{1+\exp\left(s_j\left(x_i^{(n)} - u_{ji}\right)\right)} - \frac{1}{1+\exp\left(s_j\left(x_i^{(n)} - l_{ji}\right)\right)}\right) \quad , \quad (2)$$

where l_{ji} and u_{ji} are the lower and the upper bound of the validity range of submodel j in dimension i. The parameter s_j determines the slope of the border of the validity range. Its influence on v_j is illustrated in Fig. 3.

For small s_j the slope of the border is more flat. The higher s_j gets, the steeper is the slope of the border. In this way, the transition between the submodels can be controlled. If there are smoothness assumptions about the target function one can choose a lower value for s_j.

To update the model parameter the EM algorithm is used. In the expectation step, the validity values are integrated into the computation of the posterior probability $h_j^{(n)}$ of selecting submodel j for input vector $\boldsymbol{x}^{(n)}$:

$$h_j^{(n)} = \frac{v_j\left(\boldsymbol{x}^{(n)}\right) P\left(z_j^{(n)} \left| \boldsymbol{x}^{(n)}, \theta_g\right.\right) P\left(y^{(n)} \left| \boldsymbol{x}^{(n)}, \theta_j\right.\right)}{\sum_{k=1}^{M} v_k\left(\boldsymbol{x}^{(n)}\right) P\left(z_k^{(n)} \left| \boldsymbol{x}^{(n)}, \theta_g\right.\right) P\left(y^{(n)} \left| \boldsymbol{x}^{(n)}, \theta_k\right.\right)} \quad . \quad (3)$$

This enforces the gate to reduce the weights of submodel outputs if the input vectors are located outside their domains. The particular amount of weight decrease depends on the value of v_j.

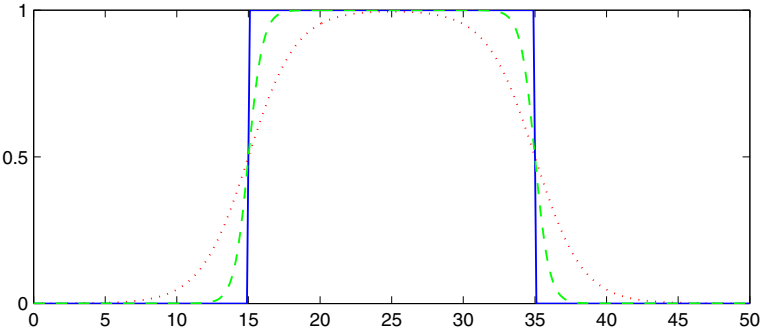

Fig. 3. The figure shows several validity ranges with different values of s: $s = 0.8$ (dottet line), $s = 2$ (dashed line), and $s = 100$ (solid line)

In the maximization step, the log likelihood function

$$L = \sum_{n=1}^{N} \sum_{j=1}^{M} h_j^{(n)} \log \left(P\left(z_j^{(n)} \left| \boldsymbol{x}^{(n)}, \theta_g \right. \right) P\left(y^{(n)} \left| \boldsymbol{x}^{(n)}, \theta_j \right. \right) \right) \tag{4}$$

is to be maximized with respect to the parameters of the gate and of the data-driven submodels.

4 Real-World Application

The application addresses the simulation of the electrical energy flow in the powertrain of a hybrid electric vehicle. Four distinct driving modes can be defined by the available expert knowledge: pure electric drive mode, hybrid drive mode, brake mode, and drag mode. Dependent on the current drive mode electrical energy is used in several different ways. In pure electric drive mode and hybrid drive mode energy is provided by the battery to drive the electric motor. In brake mode and drag mode the electric motor is operating as a generator to recuperate the kinetic energy to be used for charging the battery. Domain experts designed specific models for each mode. These models represent complementary information sources since they are defined for different regions of the input space with each model providing information for different mutually exclusive driving modes. Furthermore, the battery must maintain certain chemical limits. These limits determine the maximum charge and discharge capabilities of the battery dependent on its state of charge and temperature.

The data set is randomly divided into a training data set (80% of the data) and a test data set (20% of the data). The overall experiment is performed ten times and the results are averaged. The following models were compared: an HME, an ME, a multi-layer perceptron (MLP), and an ensemble of MLPs. The HME model uses four expert models. Two characteristic maps and a mathematical model represent the pure electric drive mode, brake, and drag mode. However,

since there is no model provided for the hybrid drive mode a two-layer MLP with 5 input units, 6 hidden units and one output unit was learned. Each mode has different input features. As gate, an MLP with 4 hidden units was applied. For each knowledge-based model j a validity function v_j is defined by the the domain experts. For the data-driven model no validity function is given. Instead, it is assumed to be valid in the entire input space.

The ME consists of 4 MLPs with 6 hidden units and as gate an MLP with 5 hidden units was used. The single MLP comprises 14 hidden units. In the ensemble 10 members were combined. All members have the same architecture, i.e. MLPs with a single hidden layer of 8 hidden units. The ensemble is generated using K-fold cross-validation, where K is the number of ensemble members. The output of the ensemble is computed as follows:

$$y_{Ens}\left(x^{(n)}\right) = \frac{1}{K}\sum_{j=1}^{K} y_j\left(x^{(n)}\right) \ , \tag{5}$$

where $y_j\left(x^{(n)}\right)$ is the output of the j ensemble member. We used the mean absolute error to compare the perfomance of the models:

$$e = \frac{1}{N}\sum_{n=1}^{N}\left|y^{(n)} - f\left(\boldsymbol{x}^{(n)}\right)\right| \ . \tag{6}$$

Table 1 summarizes the results. The HME achieves superior performance due to the incorporation of available information sources. Fig. 4 shows the outputs of the gate model (the activation of the submodels) of the HME. In most cases, the gate selects only one submodel for each input vector. This behaviour is consistent with the knowledge of the domain expert that the submodels were defined for different mutually exclusive modes. The ME model was not able to identify the driving modes and dividing the input space in a technically non-plausible way. This is illustrated in Fig. 5. The overall output is composed of the outputs of the submodels.

The chemical battery limits are violated by all models, except the HME, since they predict energy flows that cannot be provided by the battery. Some violations of the limits are shown in Fig. 6 of (a) the MLP, (b) the ME, and (c) the ensemble. The necessary information about these limits is not contained

Table 1. Mean absolute error for the hybrid electric vehicle data set

Model	Mean absolute error	
	training	testing
HME	1.82	1.84
ME	2.57	2.71
MLP	2.05	2.11
Ensemble	1.97	2.03

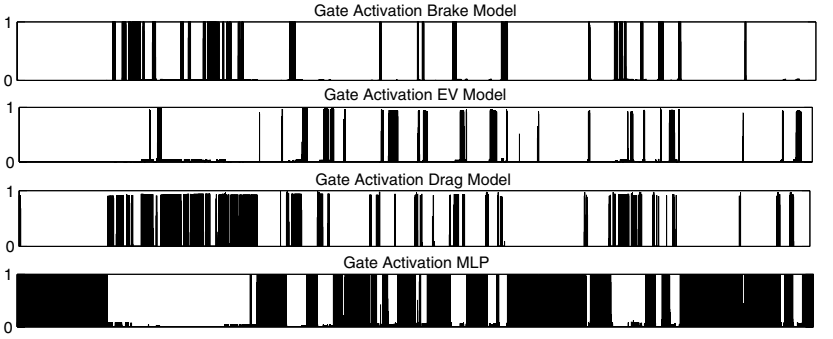

Fig. 4. The figure shows the activations of the different submodels by the gate of the HME model

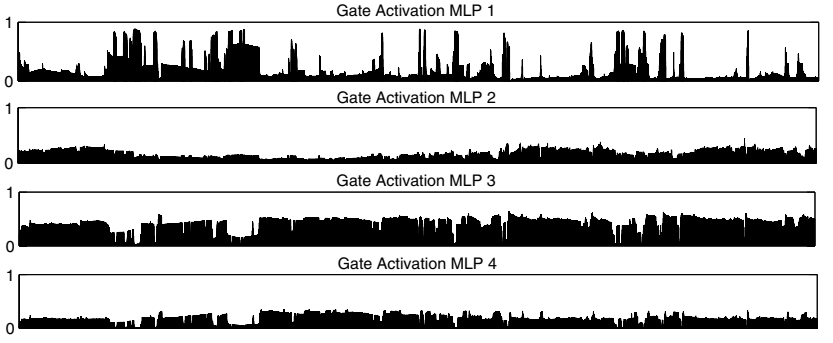

Fig. 5. The figure shows the activations of the different submodels by the gate of the ME model for the same data as shown in Fig 4

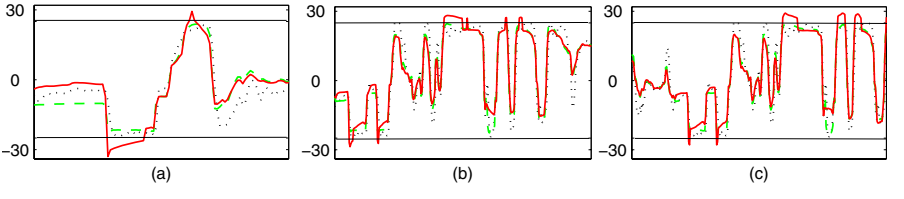

Fig. 6. The figures (a)-(c) show examples of violations of the chemical battery limits (depicted as horizontal lines) of (a) the MLP (solid line), (b) the ME (solid line), and (c) the ensemble (solid line). The target values for the energy flow and the outputs of the HME are depicted as dotted and dashed lines.

Table 2. Responsibilities of the mode models for data of the corresponding driving mode

HME Model	Driving mode (in %)			
	brake	pure electric drive	drag	hybrid
HME	97	94	92	96

Table 3. Mean absolute error for different sizes of the training data set T for the hybrid electric vehicle data set

Model	Mean absolute error				
	T	$T/2$	$T/4$	$T/8$	$T/16$
HME	1.82	1.81	1.83	1.86	1.90
ME	2.57	2.61	2.68	2.82	3.10
MLP	2.05	2.11	2.24	2.39	2.63
Ensemble	1.97	2.03	2.10	2.19	2.34

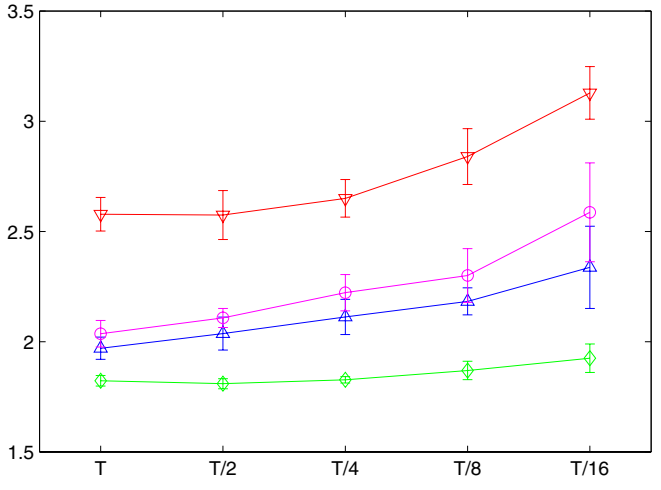

Fig. 7. The figure shows the predictive error of the models for different sizes of the training data set T. The HME model (square) has a slight increasing error for small training data set sizes. If the size of the training data set gets smaller the error of the ME model (downward-pointing triangle), the MLP (circle), and the ensemble (upward-pointing triangle) increases fast.

in the training data, but it is implicitly contained in the given knowledge-based models.

For the HME model Table 2 shows the distribution of the responsibilities of the mode models for data of the corresponding driving modes. The values indicate that the mode models are correctly assigned to the partitions of the driving modes.

An additional advantage of incorporating available knowledge is that fewer training data are required. In Table 3 and Fig. 7 the results for different sizes of the training data sets are shown. The smaller the training data set size the less robust are the results of the purely data-driven models. The results indicate that the HME model requires fewer training data compared to other regression methods in order to achieve a good predictive performance. This is useful if few training data are available.

5 Conclusions

By applying the proposed ensemble learning model it is possible to fuse information from multiple sources represented by knowledge-based models. Data-driven submodels are used to complement these models with respect to the coverage of the input space. To be able to integrate given knowledge-based models into the process of simultaneously training the data-driven submodels and a gate model it is crucial to incorporate the validity ranges of the knowledge-based models. The integration of knowledge-based models does not only lead to a superior performance but also results in an improved plausibility and reliability of the proposed model compared to the other models. Furthermore, the HME benefits from the additional information provided by the knowledge-based models as shown in the application example.

References

1. Freund, Y., Schapire, R.E.: Decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences 55(1), 119–139 (1997)
2. Jacobs, R.A., Jordan, M.I., Nowlan, S.J., Hinton, G.E.: Adaptive mixtures of local experts. Neural Computation 3, 79–87 (1991)
3. Perrone, M.P.: Improving Regression Estimation: Averaging Methods for Variance Reduction with Extensions to General Convex Measure Optmization. PhD thesis, Brown University (1993)
4. Dasarathy, B.V.: Information fusion - what, where, why, when, and how? Information Fusion 2(2), 75–76 (2001)
5. Durrant-Whyte, H.F.: Sensor models and multisensor integration. International Journal of Robotics Research 7(6), 97–113 (1988)
6. Bengio, S., Marcel, C., Marcel, S., Mariéthoz, J.: Confidence measures for multimodal identity verification. Information Fusion 3(4), 267–276 (2002)
7. Bloch, I.: Information Fusion in Signal and Image Processing: Major Probabilistic and Non-Probabilistic Numerical Approaches. John Wiley & Sons Inc., Chichester (2008)
8. Jordan, M.I., Jacobs, R.A.: Hierarchical mixtures of experts and the EM algorithm. Neural Computation 6(2), 181–214 (1994)
9. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society 39(1), 1–38 (1977)