Abstract - If knowledge such as classification rules are extracted from sample data in a distributed way, it may be necessary to combine or fuse these rules. In a conventional approach this would typically be done either by combining the classifiers’ outputs (e.g., in form of a classifier ensemble) or by combining the sets of classification rules (e.g., by weighting them individually). In this paper, we introduce a new way of fusing classifiers at the level of parameters of classification rules. This technique is based on the use of probabilistic generative classifiers using multinomial distributions for categorical input dimensions and multivariate normal distributions for the continuous ones. That means, we have distributions such as Dirichlet or normal-Wishart distributions over parameters of the classifier. We refer to these distributions as hyper distributions or second-order distributions. We show that fusing two (or more) classifiers can be done by multiplying the hyper distributions of the parameters and derive simple formulas for that task. Properties of this new approach are demonstrated with a few experiments. The main advantage of this fusion approach is that the hyper distributions are retained throughout the fusion process. Thus, the fused components may, for example, be used in subsequent training steps (online training).

Key Words: Knowledge fusion, classifier fusion, probabilistic classifier, generative classifier, data mining, Bayesian techniques.

I. INTRODUCTION

In various machine learning applications, the task of extracting knowledge (e.g., classification rules) from sample data is divided into a number of subtasks. Typical examples are smart sensor networks, robot teams, or software agents that learn locally in their environment. At some point, there is necessity to fuse or to combine the knowledge that is now “contained” in a number of classifiers in order to apply it to new data. An application in the field of distributed intrusion detection in computer networks is described. Therefore, it is impossible to exchange the raw data because of a limited communication bandwidth. Also, a central unit would constitute a single point of failure. In other data mining applications, knowledge extraction is split into subtasks due to memory or runtime limitations. Again, locally extracted knowledge must be consolidated later and quite often, the communication overhead should be low.

In this paper, we focus on classification problems and assume that probabilistic generative classifiers are used to solve these problems. Probabilistic classifiers provide outputs that can be interpreted as conditional probabilities as they model the conditional distribution of classes given an input sample. Generative classifiers aim at modeling the processes from which the sample data are assumed to originate. Probabilistic generative classifiers are usually based on Bayes’ theorem. Advantages are, for example, that

- The class posterior probabilities \( p(c/x) \) are very useful to weight single decisions when several classifiers are combined, for example, in form of ensembles,
- A rejection criterion could easily be defined which allows to refuse a decision if none of the class posteriors reaches a prespecified threshold, or
- In dynamic environments it is possible to detect novel situations, for example, data that originate from new processes that did not exist when the initial training data were collected.

All together, it depends on the type of application whether such classifiers can successfully be applied. Which possibilities do we have to fuse or to combine probabilistic generative classifiers.

II. RELATED WORK

If we talk about “knowledge fusion” we have to be much more precise to what kind of knowledge we refer. Fusion can take place at various levels or categories:

1. Data (e.g., sensor measurements or observations) or information extracted from data can be fused to come to more certain conclusions, for instance.
2. Models or parts of models trained from sample data or information can be fused if the models
were constructed in a distributed fashion.

3. The outputs of models can be fused, for example, to get more certain decisions or—as in the case of temporal and spatial data mining—to derive conclusions for certain points in space and time.

Interestingly, the term “Bayesian knowledge fusion” (which we also claim for our work) is often associated with category one. Several variants can be found in the literature, while the most interesting ones are sequential Bayesian estimation techniques or the fusion of several likelihood functions as in the case of the independent likelihood pool approach. More details on this technique which is quite distinct from ours as it addresses the first order and not the second-order distributions can be found; applications in the fields of multimedia, robotics, or target detection are outlined.

2.1 Disadvantages Of Existing System:

The challenges at these classifiers are more likely to over fit to sample data as the (effective) number of parameters is typically quite high, or the classification performance is sometimes worse if data do not (at least nearly) meet the distribution assumptions.

III. PROPOSED SYSTEM

We focus on classification problems and assume that probabilistic generative classifiers are used to solve these problems. Probabilistic classifiers provide outputs that can be interpreted as conditional probabilities as they model the conditional distribution of classes given an input sample. Generative classifiers aim at modeling the processes from which the sample data are assumed to originate. Probabilistic generative classifiers are usually based on Bayes’ theorem.

Advantages Of Proposed System:

The class posterior probabilities \( p(c / x) \) are very useful to weight single decisions when several classifiers are combined, for example, in form of ensembles,

- A rejection criterion could easily be defined which allows to refuse a decision if none of the class posteriors reaches a prespecified threshold, or
- In dynamic environments it is possible to detect novel situations, for example, data that originate from new processes that did not exist when the initial training data were collected.

IV. METHODOLOGICAL FOUNDATIONS

Probabilistic Generative Classifier CMM

In general, a classifier is a function mapping an input value \( x \) to an output class \( c \in \{1, \ldots, C\} \) of \( C \) possible classes. A probabilistic classifier takes the form \( p(c / x) \) which denotes the probability for class \( c \) given an input sample \( x \). The classifier is split into \( C \) parts, one for each class. \( p(c) \) is a multinomial distribution specifying the prior probability of class \( c \), the conditional densities \( p(x \mid c / j) \) are called components of the classifier, and \( p(j \mid c) \) is another multinomial distribution whose parameters \( c_j \) are called mixture coefficients and weight the components in their respective part of the mixture model. The overall classifier, which we call a classifier based on a mixture model (CMM), consists of \( J = PC \) components each of which is described by a (usually multivariate) distribution \( p(x \mid c / j) \). As the input data can have both, categorical and continuous dimensions, the distributions \( p(x \mid c / j) \) must be chosen in a way such that both cases can be handled by the classifier. To achieve this, we arrange the dimensions without loss of generality in a way such that the first \( D_{\text{cont}} \) dimensions of \( x \) are continuous and the following \( D_{\text{cat}} \) dimensions are categorical which results in a total of \( D = D_{\text{cont}} + D_{\text{cat}} \) dimensions.

Hyper distributions for CMM Parameters

So far, we presented the CMM classifier consisting of components with categorical and continuous dimensions. The parameters of these components are trained in a Bayesian way as will be explained in Section 3.3. To do this, prior distributions must be assumed for various parameters of the classifier. As the components of the classifier already consist of distributions and we now use another set of distributions to describe values of their parameters, the latter set of distributions is called hyper distributions or second-order distributions. The priors are then combined with a likelihood function based on training data (observations):

\[
\text{Posterior} \propto \text{likelihood} \cdot \text{prior}
\]

If we ignore the normalizing coefficient for now. From the posterior distribution, we finally get point estimates for the various parameters as described. The posteriors and the priors have the same functional form if we assume a specific functional form for the priors, the so-called conjugate priors. Then, a posterior could be used as prior again in a sequential training approach (e.g., for online training). The use of conjugate priors is the standard approach in Bayesian learning as various kinds of prior knowledge can be expressed (including the case...
of absence of prior knowledge with no informative priors). Otherwise, it would not be possible to derive fusion formulas in an elegant way.

VI Training Algorithm

The VI algorithm consists of two steps denoted by E and M which is executed in an alternating fashion until a convergence criterion is met. This can, for example, be a fixed number of steps or a condition which expresses that the classifier does not change significantly in a number of subsequent steps.

Point Estimates

When the training algorithm has converged, the classifier’s parameters are derived from the hyper distributions by means of a point estimate. A quadratic loss function will be minimized if we choose the expectations of the hyper distributions

V. A NEW CLASSIFIER FUSION TECHNIQUE

Our fusion mechanism uses the hyper distributions obtained in the VI training process. Doing so, we retain these hyper distributions throughout the fusion process which has several advantages over a simple linear combination of CMM parameters. For example, the variance of the hyper distributions can still be used to assess the uncertainty of the fused classifier’s parameters. Basically, two CMM can be used by identifying corresponding hyper distributions (e.g., those modeling corresponding processes) and multiplying these hyper distributions. If for a component of the first classifier no corresponding component of the second classifier exists (or vice versa), these components are simply combined in the resulting classifier as sketched. That is, the union of these component sets is built and the mixture coefficients are adapted accordingly.

The classifier resulting from all fusion and combinations steps is also called overall classifier in the remainder of this paper. Algorithm gives a very informal overview of the proposed algorithm. One basic assumption throughout this section is that all classifiers to be merged have the same input space, in particular the same number of categorical and continuous dimensions.

Basic Idea for Fusion

If two classifiers model similar processes they are likely to contain many similar components. We now want to detect such a situation to fuse all pairs of similar components. When two CMM are trained separately, each with a distinct part of the training data, we have two likelihood functions derived from the two sets of training data and two prior distributions. We now assume that the two priors are equal because in both cases we make use of the same prior knowledge or want to express the same amount of uncertainty about the parameters we want to estimate, for instance. Thus, this is not a harsh restriction at all. Nevertheless, this leaves us with two posterior distributions.

A Similarity Measure for Hyper distributions

To determine whether two components should be fused or not we use a threshold #H 2 =0; 1. Higher values of this threshold mean that only very similar components are fused while low values lead to the fusion of nearly all components. Some examples of applying the similarity measure are given. Instead of using H to determine whether or not two components of a CMM should be fused it would also be possible to implement different approaches. For example, in some applications it could be useful to concentrate on the distance between the two centers of the normal distributions and ignore the covariance matrices. This shall not be discussed here further as it has to be decided based on the specific application. The similarity measure _H directly operates on the normal and multinomial distributions of the classifier. Theoretically, it would also be possible to compute the Hellinger distance of the hyper distributions to evaluate the similarity of components but in that case the integral in (39) could not be computed in a closed form. For two CMM classifiers, we can now decide which components we want to fuse.

Fusion of Normal-Wishart Distributions

According to Section, the conjugate prior distribution that must be used to estimate the parameters of a multivariate normal distribution is a normal-Wishart distribution. To fuse two normal-Wishart distributions we multiply their density functions and divide the result by the density of the prior. The density of a normal-Wishart distribution has been shown. Basically, it can be seen as a product of a normalization factor, a Gaussian factor, and a Wishart factor.
VI. EXPERIMENTAL RESULTS

In this section, we first give some examples concerning values of the similarity threshold mentioned. Then we apply the developed fusion techniques for CMM classifiers to artificial and real-world data sets.

Suitable Fusion Parameters

In Section, we presented a similarity measure $H$ that can be used to decide whether or not two components should be fused. In an application the question arises, which value should be chosen for the fusion threshold $#H$ to guarantee that only similar components are actually fused. This section gives some examples for both continuous and categorical dimensions to make the reader familiar with the characteristics of $H$. The actual value of $#H$, however, has always to be determined based on the specific application. Similarity values for two-dimensional Gaussians whose centers differ and whose covariance matrices always equal the identity matrix. The displayed circles correspond to the surface of constant density where the Mahalanobis distance of the points on the circles to the center of the respective distribution is 1. As expected, the greater the difference between the centers of the normal distributions, the lower is their similarity. In Fig. 7b, the difference between the centers is 0.25 in the first dimension and 0 in the second dimension. This evaluates to a similarity of the two Gaussians of approximately 0.91. In Fig. 7b, the difference between the centers is 0.25 in both dimensions which leads to a similarity value of 0.88. Finally, the difference is increased to 0.5 in both dimensions and the similarity lowers to a value of 0.75. If the covariance matrix of the two Gaussians is varied instead of the centers, similar results are obtained. Shows three pairs of two-dimensional normal distributions whose covariance matrices differ. The covariance matrix of one of the distributions always equals the identity matrix. The other distribution has the covariance matrices.

Finally, we want to emphasize that a similarity threshold for fusion has to be adjusted by the user depending on the application. It is influenced by parameters such as the type of the dimensions (categorical or continuous), the number of dimensions, or the number of categories in the case of categorical dimensions. If there are many dimensions the fusion threshold should typically be lower as components need to have a greater overlap in order to have a larger similarity value. For few dimensions, the threshold can have a higher value. A similar rule applies to the number of categories.

Artificial Data Set

In this section, we apply the new fusion method to an artificial data set because the results can easily be depicted to get a visual impression of fusion results (combination does not occur in the examples shown here). The data set used here is the Clouds data set taken. It consists of 5,000 samples that were generated from four Gaussian processes: one belongs to the first, green class (whose samples are depicted by little circles) and three belong to the second, blue class (whose samples are depicted by little plus signs). The data were normalized using a z-transformation and we conduct a fivefold cross validation, i.e., we use 4/5 of the samples as training data and 1/5 of the data to test the resulting classifier. Instead of training just one classifier with all available training data (referred to as complete-data classifier in the following) we split the training data again into two equal-sized parts, train two classifiers, and fuse them afterward to obtain the overall classifier of the respective fold. Fig. 9 shows the results of the last fold of the cross validation. Display the two classifiers together with their respective part of the training data. The ellipses correspond to the Gaussian components of the CMM and contain the area whose Mahalanobis distance (cf. (5)) to the center of the component is less or equal to 1. The centers of the components are marked with small symbols. The dashed green ellipses mark components that belong to the first class and dotted blue ellipses mark components that belong to the second class. The solid black line marks the decision boundary.

VII. CONCLUSION

In this paper, we presented a novel technique to fuse two probabilistic generative classifiers (CMM) into one. A CMM consists of several components each of which may in turn consist of one multivariate normal distribution modeling continuous dimensions of the input space and multiple multinomial distributions, one for each categorical dimension of the input space. To identify components of two classifiers that shall be fused, we suggested a similarity measure that operates on the distributions of the classifier. The actual fusion of two components works one level higher on the hyper distributions which are the result of the Bayesian training of a CMM using the VI (variational Bayesian inference) algorithm. We derived formulas to fuse both Dirichlet and normal-Wishart distributions which are the conjugate prior distributions of the multinomial and normal distributions of a CMM in a very elegant way (see Table 1). Experiments with some benchmark data
sets outlined the properties of this new knowledge fusion approach.

The extension of our knowledge fusion approach to more than two CMM classifiers is straightforward as it is possible to apply the technique iteratively. It will certainly be possible to use the same parameter values (fusion threshold) for all single fusions. While being trivial from a technical point of view, the actual advantages for real applications have still to be pointed out in our future work. If the number of classifiers is known in advance it would also be possible to modify the fusion formulas accordingly. In our future work, we will also generalize the approach to other distributions, in particular members of the exponential family of distributions and investigate how different prior distributions can be handled. We also have to find a more intuitive way to parameterize the fusion threshold and we will investigate the weighting of categorical and continuous dimensions in more detail. The proposed techniques could be used in the field of distributed data mining, where data sets have to be split to cope with huge amounts of data and where the communication costs have to be low. It is also possible to use them in distributed environments where data are locally processed as they arise locally (e.g., in smart sensor networks). A very specific application has been proposed by the authors of this paper: collaborative learning, where intelligent technical systems learn from each other by exchanging locally learned rules that are potentially useful for others. In this intrusion detection application, which has already been mentioned in the introduction, classifiers are fused and not combined to keep the number of model components as low as possible. This allows for an effective combination of generative models with anomaly detection techniques to detect new kinds of attacks.

REFERENCES


