"A Globally-Variant Locally-Constant Model for Fusion of Labels from Multiple Diverse Experts Without Using Reference Labels"

Authors: Kartik Audhkhasi, Shrikanth S. Narayanan

IEEE Transactions on Pattern Analysis and Machine Intelligence, Accepted: June 12, 2012

DOI: 10.1109/TPAMI.2012.139

Summary of Novel Ideas in the Paper (different from Abstract):

Ensembles of machine experts, from simple linear classifiers to complex hidden Markov models, have out-performed single experts across many applications. Likewise, ensembles have been central to computing with human experts e.g., for data annotation. This widespread use of ensembles, albeit largely heuristic, is motivated by their better generalization and robustness to ambiguity in the production, representation, and processing of information.

Optimal fusion of labels from multiple experts is critical to exploit their diversity. Simple plurality, while popular, however gives equal importance to labels from all experts who may not be equally reliable and consistent across the data set. Previous works often assume constant reliability. We present a general Bayesian model based on the consideration that in real-world data, expert reliability is variable over the complete feature space but constant over clusters of homogeneous instances. This model jointly learns a classifier and expert reliability parameters without using reference labels through the Expectation-Maximization algorithm. Benefits of this model are shown through experiments on data from the UCI Machine Learning Repository and on emotional speech classification data sets. A metric based on the Jensen-Shannon divergence shows that the proposed model gives greater benefit when expert reliability is more variable over the feature space.

A Globally-Variant Locally-Constant Model for Fusion of Labels from Multiple Diverse Experts Without Using Reference Labels

Kartik Audhkhasi, Shrikanth S. Narayanan

Signal Analysis and Interpretation Lab (SAIL)

Electrical Engineering Department

University of Southern California, Los Angeles, USA

Email: audhkhas@usc.edu, shri@sipi.usc.edu

Digital Object Indentifier 10.1109/TPAMI.2012.139

Abstract

It has been shown that fusion of categorical labels from multiple experts – humans or machine classifiers – improves the accuracy and generalizability of the overall classification system. Simple plurality is a popular technique for performing this fusion, but it gives equal importance to labels from all experts who may not be equally reliable and consistent across the data set. Estimation of expert reliability without knowing the reference labels is however a challenging problem. Most previous works deal with these challenges by modeling expert reliability as constant over the entire data (feature) space. This paper presents a model based on the consideration that in dealing with real-world data, expert reliability is variable over the complete feature space but constant over local clusters of homogeneous instances. This model jointly learns a classifier and expert reliability parameters without assuming the knowledge of reference labels using the Expectation-Maximization algorithm. Classification experiments on simulated data, data from the UCI Machine Learning Repository and two emotional speech classification datasets show the benefits of the proposed model. Using a metric based on the Jensen-Shannon divergence, we empirically show that the proposed model gives greater benefit for data sets where expert reliability is highly variable over the feature space.

Index Terms: Multiple Diverse Experts, Label Fusion, Label Reliability, Expectation-Maximization Algorithm, Human Annotation, Emotion Recognition

I. Introduction

Conventional supervised pattern classification assumes the availability of a reference label for each training instance based on two implicit assumptions. First, the set of classes is assumed to be unambiguous and crisply defined. This may not hold in many real-world scenarios due to the inherent non-categorical and ambiguous nature of the phenomena of interest in both their manifestation and human processing. A classic example is emotion recognition from speech. It is well known that the expression and perception of natural human emotions is complex and characterized by heterogeneity [1]. For example, while the emotional content of a person's speech may appear predominantly angry, it may have different shades of anger. Furthermore, it may contain acoustic characteristics of neutrality and sadness as well. Thus discretization of the emotional description into one of a few categories such as angry or sad is only an approximation to the underlying continuum. The second common assumption behind the availability of a reference label is that the labeling process is reliable, i.e., the correct class label has been assigned to each instance. This assumption is often unrealistic since even expert labelers typically do not possess complete knowledge of the classes. For example, a human expert labeling

emotional speech is biased by his prior experience about the acoustic characteristics of various emotions, which may not be in consonance with the speech clips being appraised (labeled). Machine classifiers also generate labels for unseen data instances using the instance-to-label mapping learned from a training corpus, which may not generalize to the test data. This results in classification errors. Finally, in some situations, obtaining the true label can be expensive, time-consuming, or even dangerous. For example, in the medical domain, labeling a tissue as malignant or otherwise can be done through biopsy, which is not only an expensive procedure but invasive [2] as well. Supervised training of spoken language systems (e.g. automatic speech recognizers) typically requires professional transcription of large amounts of speech data, which is both time consuming and expensive [3].

A simple strategy for dealing with the above issues is to get each instance labeled by multiple potentially imperfect experts (a generic term for a human labeler or machine classifier). This is followed by a simple plurality fusion, wherein the class label with most votes is deemed the reference. Consider R experts and a label set with K classes, denoted by $\{1, ..., K\}$. Let y^j denote the label assigned by the j^{th} expert and y_k^j its 1-in-K encoding, i.e., $y_k^j = 1$ if $y^j = k$, and 0 otherwise. Formally, simple plurality uses the following decision rule:

$$\hat{y}_{PLU} = \arg\max_k \sum_{j=1}^{R} y_k^j \tag{1}$$

Thus, it gives equal weight to the labels from all experts. However, since the reliability of different experts can be variable and also data dependent, it is reasonable to emphasize a more reliable expert's judgment while making the overall decision. But computing expert reliability in many real world data labeling problems is challenging and can be exacerbated by the hidden or ambiguous nature of the true class labels in many cases. A simple but powerful approach to this problem based on the Expectation-Maximization (EM) algorithm was proposed by Dawid and Skene [4], and later by Smyth et al. [5], as described in Section II. In this model, the Bayes optimal maximum a-posteriori (MAP) decision rule for combining labels from R experts becomes a weighted sum of their 1-in-K encoding.

One of the limitations of the model in [4], [5] is that a classifier has to be learnt separately from the estimation of the labeling parameters, making the overall estimation sub-optimal. To overcome this difficulty, Raykar et al. [2] proposed an extension by explicitly incorporating a classifier linking the feature vector and the hidden reference label. The accuracy of this model

over a variety of datasets was shown to be better than a classifier learned using the labels obtained from simple plurality fusion and the model in [4], [5].

We note however that, in the above models, expert reliability is assumed to be constant over all data instances. However, in real-world scenarios, this reliability varies from one instance to another, as illustrated by means of examples in the next section. Furthermore, instances close to each other in the feature space tend to be labeled with similar reliability. These two ideas form the basis for the new expert labeling model proposed in this paper. In this model, the feature variability is captured by a generative model; we consider a feature space generated using a Gaussian Mixture Model (GMM). The hidden reference label is assumed to be generated from a given feature vector using the multinomial logistic regression or maximum entropy (MaxEnt) model. As will become apparent later, any classifier that can be trained with soft labels can be used. Each expert's reliability is assumed to be constant only over each mixture component, as opposed to the entire feature space.

A preliminary version of this model was presented at Interspeech-2010 [6]. The present paper develops the model in a Bayesian framework for the multi-class case and analyzes the different models both theoretically and experimentally. Detailed experiments are presented using a variety of data sets that include simulated data and standard databases from the UCI Machine Learning Repository. Finally, results on tasks of classifying four emotional categories, as well as emotional valence, activation and intensity from speech are reported. While the experts in the case of emotional speech tasks refer to human evaluators, they are machine classifiers in the case of the UCI databases. A review of prior work in this domain is presented in the next section, followed by a description of our model in Section III. Details on experiments conducted on simulated and real world databases are described in Section IV. Subsection IV-D attempts to explain the observed benefit of the proposed model. The conclusions of the paper and some directions of future work are presented in Section V.

II. Prior Work

Fig. 1 shows the Bayesian network for one of the first models proposed for this problem, due to Dawid and Skene [4] and Smyth et al. [5]. Let y be a K-valued random variable that represents the unobserved reference label for a given training example. It is assumed that each of the R experts is characterized by a $K \times K$ reliability matrix $\mathbf{A}^j$, $j \in \{1, ..., R\}$. When asked

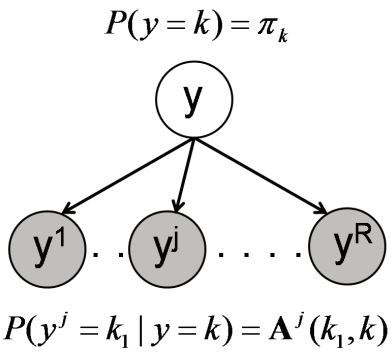

Fig. 1. Bayesian network for the model presented in [4], [5]. Shaded and unshaded nodes represent observed and unobserved random variables respectively.

to give a label corresponding to the true label $y = k$, the j^{th} expert samples from the K-valued conditional distribution $\{\mathbf{A}^j(k_1, k)\}_{k_1=1}^K$. $\mathbf{A}^j(k_1, k)$ is the probability that the j^{th} expert confuses class k_1 for class k. Given a training corpus, i.e., N independent and identically distributed (IID) R-tuples of labels from the R experts, the learning task is to estimate the prior probability distribution of y and $\{\mathbf{A}^1, ..., \mathbf{A}^R\}$. The authors find maximum likelihood (ML) estimates of these parameters by using the EM algorithm [7]. Once the parameter estimation is complete, the true label y can be inferred given observed noisy labels $\{y^1, ..., y^R\}$ as follows:

$$\hat{y}_{MAP} = \arg\max_k \log P(y = k|y^1, ..., y^R) = \arg\max_k \left[\log \pi_k + \sum_{j=1}^{R} \sum_{k_1=1}^{K} y_{k_1}^j \log \mathbf{A}^j(k_1, k) \right] \quad (2)$$

This decision rule is a weighted simple plurality, where the label from the more reliable expert is given greater weight. Within the emotion recognition community, there have been similar intuitively inspired efforts to incorporate evaluator reliability during label fusion [8]. Prop. 1 states a sufficient condition for the equivalence of this decision rule and simple plurality.

Proposition 1. *If the prior probability distribution of y is uniform ($\pi_k = \frac{1}{K}\ \forall k \in \{1, ..., K\}$) and all expert reliability matrices are equal ($\mathbf{A}^j = \mathbf{A}\ \forall j \in \{1, ..., R\}$) with the following values:*

$$\mathbf{A}(k, k) = \frac{\alpha}{(\alpha + K - 1)} \quad \forall k \in \{1, ..., K\} \quad (3)$$

$$\mathbf{A}(t, k) = \frac{1}{(\alpha + K - 1)} \quad \forall t \neq k \quad (4)$$

where $\alpha \in \mathbb{R}^+ - \{1\}$, then Eq. 2 reduces to the simple plurality rule in Eq. 1.

Proof. The pairwise discriminant functions for the two decision rules in Eq. 1 and Eq. 2 can

be written as (for $k \neq t$):

$$h(k,t) = \sum_{j=1}^{R}(y_k^j - y_t^j) \quad \text{and} \quad g(k,t) = \log\frac{\pi_k}{\pi_t} + \sum_{j=1}^{R}\sum_{k_1=1}^{K} y_{k_1}^j \log\frac{\mathbf{A}^j(k_1,k)}{\mathbf{A}^j(k_1,t)} \tag{5}$$

The two decision rules can now be written in terms of the pairwise discriminant functions:

$$\hat{y}_{PLU} = k \iff h(k,t) \geq 0 \quad \forall t \neq k \quad \text{and} \quad \hat{y}_{MAP} = k \iff g(k,t) \geq 0 \quad \forall t \neq k \tag{6}$$

Equality of the two pairwise discriminant functions is a sufficient condition for the $\hat{y}_{PLU}$ to be equal to $\hat{y}_{MAP}$. Next, we note that $g(k,t)$ can be written as follows:

$$g(k,t) = \log\frac{\pi_k}{\pi_t} + \sum_{j=1}^{R} y_k^j \log\frac{\mathbf{A}^j(k,k)}{\mathbf{A}^j(k,t)} + \sum_{j=1}^{R} y_t^j \log\frac{\mathbf{A}^j(t,k)}{\mathbf{A}^j(t,t)} + \sum_{j=1}^{R}\sum_{k_1 \neq k,t} y_{k_1}^j \log\frac{\mathbf{A}^j(k_1,k)}{\mathbf{A}^j(k_1,t)} \tag{7}$$

Comparing the above equation with $h(k,t)$, we obtain the following conditions for $g(k,t)$ to be equal to $h(k,t)$:

$$\pi_k = \pi_t; \quad \mathbf{A}^j(k,t) = \frac{1}{\alpha}\mathbf{A}^j(k,k), \quad \mathbf{A}^j(t,k) = \frac{1}{\alpha}\mathbf{A}^j(t,t) \quad \forall j$$
$$\mathbf{A}^j(k_1,k) = \mathbf{A}^j(k_1,t) \quad \forall j \text{ and } k_1 \neq k,t \tag{8}$$

where $\alpha \in \mathbb{R}^+/\{1\}$ is the base of the logarithm. Since $\mathbf{A}^j$ is singly stochastic with entries of each column adding to 1, one can find its entries using the above constraints. These turn out to be the same as in Eq. 3 and Eq. 4.

Prop. 1 requires that for each expert, the probabilities of retaining the true label are same for all values of y. In addition, the probability of making an error is constant for all choices of the true and noisy label. Finally, the reference label should be equally likely to assume any one of the K possible values. $\alpha \in [0,1)$ implies that $\mathbf{A}^j(k,k) < \mathbf{A}^j(t,k)\ \forall t \neq k$ and j, which means that all experts are adversarial and less likely than chance to retain the true label. $\alpha = 0$ denotes totally adversarial experts who always flip the true label to some incorrect label. For $\alpha \in [1,+\infty)$, $\mathbf{A}^j(k,k) > \mathbf{A}^j(t,k)$ and all experts are non-adversarial. $\alpha \to +\infty$ denotes perfect experts, who always retain the true label.

If one needs a classifier, the above model can be first used to infer the true hidden label y. Then a mapping between the given feature vector $\mathbf{x}$ and y can be learned. However, while these two learning steps are individually optimal, the overall process is not. To overcome this limitation, Raykar et al. [2] jointly learn the classifier and the expert reliability matrices (Fig. 2). As compared to Fig. 1, we observe that the generation of the hidden reference label y from $\mathbf{x}$ is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

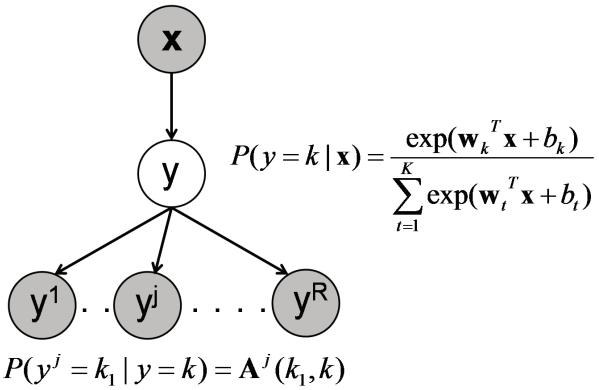

Fig. 2. Model presented by Rakyar et al. [2]. A MaxEnt classifier mapping the feature vector $\mathbf{x}$ to the true hidden label y is explicitly included in this model.

explicitly included in the model through a MaxEnt classifier. ML parameter estimation is again performed within the EM framework. The classifier is trained on soft instead of hard labels since it utilizes the posterior distribution of y estimated during the E-step. If in addition to the feature vector, noisy labels from R experts are also available, one can infer the true label y as follows:

$$\hat{y}_{MAP} = \arg\max_k P(y=k|y^1,...,y^R,\mathbf{x}) = \arg\max_k \Big[\log P(y=k|\mathbf{x}) + \log P(y^1,...,y^R|y)\Big]$$
$$= \arg\max_k \Big[\mathbf{w}_k^T\mathbf{x} + b_k + \sum_{j=1}^{R}\sum_{k_1=1}^{K} y_{k_1}^j \log \mathbf{A}^j(k_1,k)\Big] \quad (9)$$

This decision rule is very similar to Eq. 2 except for the presence of an affine function of $\mathbf{x}$ due to the MaxEnt classifier instead of the prior probability of y. Using a different classifier would modify this term. The second term however would still remain a weighted linear combination of the decisions given by the R experts.

In both the above models, each expert's reliability matrix is assumed to be constant over the entire feature space, i.e., independent of $\mathbf{x}$. In other words, these models capture the global reliability of experts. However, it is natural to expect reliability to be variable over the feature space. The primary reason for this is the fact that all instances are not equally easy to label (and for all experts). For example, Fig. 3 shows images of "4", "9" and "1" from the UCI Handwritten Pen Digits database [9]. The first row shows standard styles of writing the digits, while the next row shows non-prototypical styles. While a human or machine expert will have no difficulty in correctly recognizing the digits in the first row, the non-prototypical styles may be more easily misrecognized as some other digits. Fig. 4 illustrates the variable nature of expert reliability even further. K-means clustering (with $K=4$) was performed on the Yeast database [10] from UCI

using one of the features. Three classifiers (Logistic regression, Naive Bayes and J48 decision tree) were trained using the entire feature set, and their error rates were computed over the 4 clusters. As can be observed, the error rates are variable across the three classifiers. Moreover, there exists variability in error rate within a given type of classifier as well. Thus, modeling reliability as constant across the entire feature space is a very strict assumption.

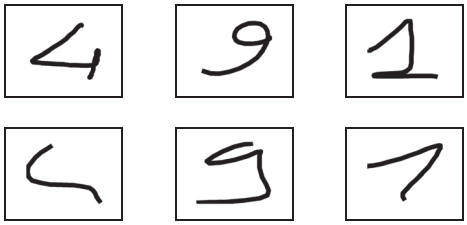

Fig. 3. Prototypical and non-prototypical shapes of three digits (4, 9 and 1) from the Handwritten Pen Digits database [9] in the first and second rows respectively.

Whitehill et al. [11] model this behavior by incorporating a difficulty parameter for each instance in addition to a measure of expert reliability. The i^{th} instance has difficulty $1/\beta_i \in \mathbb{R}^+$. $1/\beta_i \rightarrow +\infty$ implies that the instance is extremely difficult to label, while $1/\beta_i \rightarrow 0$ denotes a simple to label instance. Reliability of the j^{th} expert is governed by parameter $\alpha_j \in \mathbb{R}$, where large positive values of α_j denote a more reliable expert. The probability of retaining the true label is assumed to be a bilinear sigmoid function – $P(y_i^j = y_i) = \frac{1}{1+e^{-\alpha_j \beta_i}}$. The authors propose an EM algorithm to learn all α and β parameters, in addition to inferring a soft estimate of the true hidden label. However, given a new instance, one has to run the training over the expanded

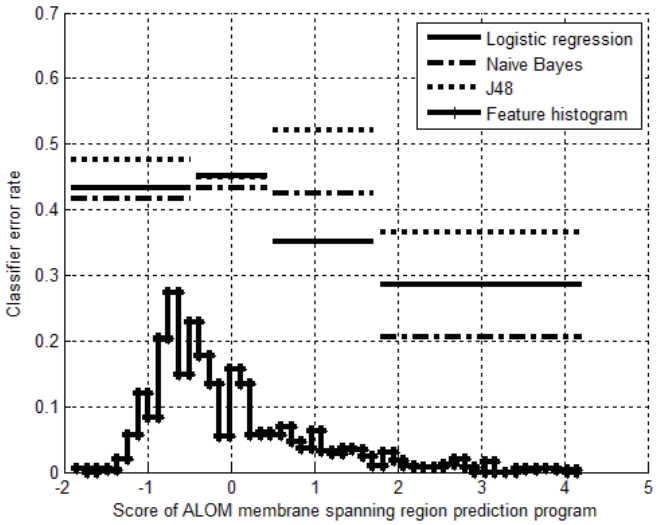

Fig. 4. Variation of error probability for three classifiers (logistic regression, Naive Bayes and J48) on the Yeast database with one of the features. The data was divided into 4 clusters using K-means and the error rates were computed over these. The feature histogram has been scaled by 2.5 for illustration purposes.

database in order to learn its difficulty and infer the hidden label, which is time consuming. Also, the number of parameters grows linearly with the number of instances, potentially leading to overfitting. In addition, a classifier is not trained jointly with the estimation of other parameters.

Another model which considers a different error probability for each instance is presented in [12]. The hidden reference label is generated from the feature vector $\mathbf{x}$ by a binary logistic regression model. The probability of retaining the true label given the feature vector is $\eta^j(\mathbf{x}) = \frac{1}{1+e^{-\mathbf{w}^{jT}\mathbf{x}-\gamma^j}}$ for the j^{th} expert. Thus, a specific form of the expert's reliability matrix (analogous to a binary symmetric channel) is adopted. Also, $\eta^j(\mathbf{x})$ is constrained to be a logistic function of $\mathbf{x}$. Each expert is characterized by the parameters $\mathbf{w}^j$ and γ^j. Apart from the above restrictions, this model is also not totally generative since it cannot generate the feature vector $\mathbf{x}$.

Our proposed model presented in the next section attempts to address the above concerns with previous models. Not only is expert reliability globally-variant and constant over clusters in the feature space, the model is truly generative as well, i.e., sampling from the model generates an instance of the feature vector and multiple noisy expert labels. This enables us to better explain the joint variability of the feature vector and labels from multiple experts. This model also does not make any restrictive assumptions about the form of the reliability matrix. We note that there has been interest recently for data processing in different communities regarding crowd-sourcing services such as Amazon Mechanical Turk and Crowd Flower. Examples range from speech/natural language processing [3], [13], computer vision [14], [15] to visualization design [16]. Most of these works require multiple experts performing some complex labeling task such as paraphrasing an article. In this paper, we will only concentrate on combination of simple categorical class labels from multiple experts.

III. The Globally-Variant Locally-Constant Model

As previously mentioned, one of the strict assumptions of the models in [4], [5], [2] is that each expert's reliability is assumed to be identical over the entire feature space. However, practical examples such as those presented in the previous section illustrate that this assumption may not hold always. Expert reliability can vary from one data instance to the next. However, imposing a different reliability matrix for each instance will lead to a huge number of parameters to be estimated. One way in which the number of parameters can be kept down to a manageable number is by requiring the reliability matrices to be constant over local clusters of instances

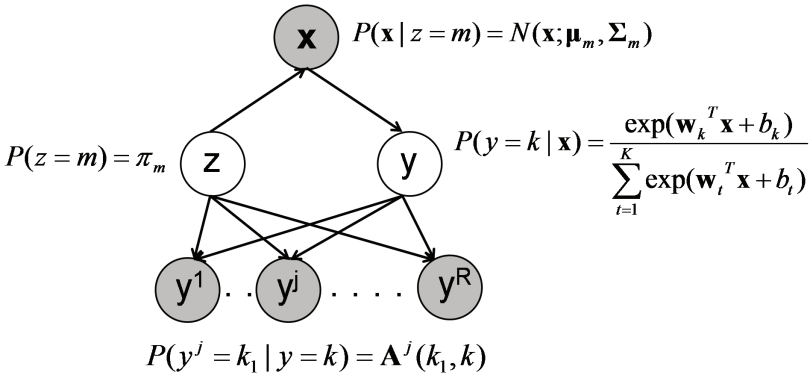

Fig. 5. The proposed data-dependent expert model. z is the hidden variable in the GMM which generates the feature vector $\mathbf{x}$.

in the feature space. This is intuitive, since we anticipate experts to have similar reliability for instances which are close to each other in the feature space. We note however that finding representative features, distance metrics to quantify feature similarity and hence clusters in the feature space, are all challenging problems. Instances close to each other in the feature space may not be perceptually close to an expert. In this paper, we assume that the given feature space organization retains the perceptual closeness of instances.

Fig. 5 shows the proposed model. The feature space distribution is modeled by a GMM. In case of a Gaussian centered at each data instance, a GMM becomes a kernel density estimator and converges to the true feature space distribution [17]. One can substitute the GMM by a mixture of discrete distributions in case the feature space is discrete. Each expert has a reliability matrix at each Gaussian in the GMM, thus modeling the data-dependent reliability while keeping the total number of parameters manageable. The generative process for $(\mathbf{x}, y^1, ..., y^R)$ is:

1) A Gaussian is selected from the M-valued distribution of z i.e., $P(z = m) = \pi_m$. If the m_0^{th} Gaussian is selected, the feature vector $\mathbf{x}$ is sampled from $\mathcal{N}(\mathbf{x}; \mu_{m_0}, \Sigma_{m_0})$.
2) The reference label is generated using the K-valued distribution implied by the MaxEnt classifier, i.e., $P(y = k|\mathbf{x}) \propto \exp(\mathbf{w}_k^T\mathbf{x} + b_k)$. Let $y = k_0$ be the sampled reference label.
3) The label for the j^{th} expert is generated by sampling the K-valued distribution in the k_0^{th} column of the reliability matrix $\mathbf{A}^j_{m_0}$, i.e., $P(y^j = k_1|y = k_0, z = m_0) = \mathbf{A}^j_{m_0}(k_1, k_0)$. y^{j_1} and y^{j_2} $(j_1 \neq j_2)$ are assumed to be independent given $y = k_0$ and $z = m_0$.

Not only does this model address issues such as labeler variability and its data dependence, it is also flexible. Increasing the number of Gaussians will lead to a finer modeling of the feature distribution and expert reliability variation. In the case of one mixture component, each expert

will have one global reliability matrix, similar to [2].

A. ML Parameter Estimation using the EM algorithm

Consider N IID training instances, each consisting of the D-dimensional feature vector $\mathbf{x}_i$ and R noisy expert labels $\{y_i^1 = k_i^1, ..., y_i^R = k_i^R\}$ ($i \in \{1, ..., N\}$). Each label y_i^j can assume K possible values, denoted by $\{1, ..., K\}$. Let the entire set of parameters to be estimated be denoted by $\Theta = \{(\pi_m, \mu_m, \Sigma_m, (\mathbf{A}_m^j)_{j=1}^R)_{m=1}^M, (\mathbf{w}_k, b_k)_{k=1}^K\}$. The observed data ($\mathcal{D}_{obs}$) log-likelihood is:

$$\begin{aligned}\log P(\mathcal{D}_{obs}|\Theta) &= \sum_{i=1}^{N} \log P(\mathbf{x}_i, (y_i^j = k_i^j)_{j=1}^R|\Theta) \\ &= \sum_{i=1}^{N} \log \Big(\sum_{m=1}^{M} \sum_{k=1}^{K} P(\mathbf{x}_i, (y_i^j = k_i^j)_{j=1}^R, y_i = k, z_i = m|\Theta) \Big) \end{aligned} \tag{10}$$

ML estimation of Θ by direct maximization of $\log P(\mathcal{D}_{obs}|\Theta)$ is mathematically intractable. Thus we resort to the EM algorithm with z_i and y_i ($i \in \{1, ..., N\}$) as the unobserved data. The complete data log-likelihood can be shown to factor as follows:

$$\begin{aligned}\log P(\mathcal{D}_{obs}, \mathcal{D}_{unobs}|\Theta) = \sum_{i=1}^{N} \log P(\mathbf{x}_i, (y_i^j = k_i^j)_{j=1}^R, y_i, z_i|\Theta) = \sum_{i=1}^{N} \Big[\sum_{m=1}^{M} z_{im} \big\{ \log \pi_m \\ + \log \mathcal{N}(\mathbf{x}_i; \mu_m, \Sigma_m) + \sum_{k=1}^{K} y_{ik} \log \sigma(\mathbf{x}_i; \mathbf{w}_k, b_k) + \sum_{k=1}^{K} \sum_{j=1}^{R} y_{ik} \log \mathbf{A}_m^j(k_i^j, k) \big\} \Big] \end{aligned} \tag{11}$$

where $\sigma(\mathbf{x}_i; \mathbf{w}_k, b_k) = \frac{\exp(\mathbf{w}_k^T \mathbf{x}_i + b_k)}{\sum_{\tilde{k}=1}^{K} \exp(\mathbf{w}_{\tilde{k}}^T \mathbf{x}_i + b_{\tilde{k}})}$, and z_{im} and y_{ik} denote the 1-in-M and 1-in-K encodings of z_i and y_i respectively. We now need to compute the posterior probability density function (PDF) of hidden variables (z_i and y_i) given the observed variables and the current parameter estimates. This can be written as follows:

$$\begin{aligned} P(z_i = m, y_i = k|\mathbf{x}_i, (y_i^j = k_i^j)_{j=1}^R) \propto P(z_i = m, y_i = k, \mathbf{x}_i, (y_i^j = k_i^j)_{j=1}^R) \\ = P(z_i = m) P(\mathbf{x}_i|z_i = m) P(y_i = k|\mathbf{x}_i, z_i = m) P((y_i^j = k_i^j)_{j=1}^R|z_i = m, \mathbf{x}_i, y_i = k) \\ \therefore \zeta_{ikm} \propto \pi_m \mathcal{N}(\mathbf{x}_i; \mu_m, \Sigma_m) \sigma(\mathbf{x}_i; \mathbf{w}_k, b_k) \prod_{j=1}^{R} \mathbf{A}_m^j(k_i^j, k) \end{aligned} \tag{12}$$

We note that $\zeta_{ikm} = \mathbb{E}\{y_{ik} z_{im}\}$. The expectation of the complete data log-likelihood with respect to the above posterior PDF additionally involves computation of the following quantities:

$$\mathbb{E}\{z_{im}\} = P(z_i = m|\mathbf{x}_i, (y_i^j = k_i^j)_{j=1}^R) = \sum_{k=1}^{K} \zeta_{ikm} = \gamma_{im} \tag{13}$$

$$\mathbb{E}\{y_{ik}\} = P(y_i = k|\mathbf{x}_i, (y_i^j = k_i^j)_{j=1}^R) = \sum_{m=1}^{M} \zeta_{ikm} = \eta_{ik} \tag{14}$$

The expectation of the complete data log-likelihood thus becomes:

$$\mathbb{E}\{\log P(\mathcal{D}_{obs}, \mathcal{D}_{unobs}|\Theta)\} = \sum_{i=1}^{N}\sum_{m=1}^{M}\Big\{\gamma_{im}\log\pi_m + \gamma_{im}\log\mathcal{N}(\mathbf{x}_i;\mu_m,\Sigma_m)\Big\} + \sum_{i=1}^{N}\sum_{k=1}^{K}\eta_{ik}\log\sigma(\mathbf{x}_i;\mathbf{w}_k,b_k) + \sum_{i=1}^{N}\sum_{m=1}^{M}\sum_{k=1}^{K}\sum_{j=1}^{R}\zeta_{ikm}\log\mathbf{A}_m^j(k_i^j,k) \quad (15)$$

The M-step consists of maximizing the above expectation with respect to $\mathbf{\Theta}$ subject to:

$$\sum_{m=1}^{M}\pi_m = 1 \quad \text{and} \quad \sum_{k_1=1}^{K}\mathbf{A}_m^j(k_1,k) = 1 \quad \forall j,m,k \quad (16)$$

Using the Lagrange multiplier method, the re-estimation equations for the GMM parameters and reliability matrices can be determined. The MaxEnt parameters can be estimated by solving the following optimization problem:

$$(\hat{\mathbf{w}}_k,\hat{b}_k) = \arg\max_{\mathbf{w}_k,b_k}\sum_{i=1}^{N}\sum_{k=1}^{K}\eta_{ik}\log\sigma(\mathbf{x}_i;\mathbf{w}_k,b_k) = \arg\min_{\mathbf{w}_k,b_k}\mathcal{E}((\mathbf{w}_k,b_k)_{k=1}^{K}) \quad (17)$$

where $\mathcal{E}$ is the negative of the MaxEnt objective function, also called cross-entropy. The above objective function is the same as for a conventional MaxEnt classifier when η_{ik} is the 1-in-K encoding of the hard label of the i^{th} instance. Since the objective function is convex, any gradient based method can be used to find the parameter estimates. Moreover, the gradient and Hessian of this objective function can be found analytically as follows [18]:

$$\nabla_{\mathbf{w}_k,b_k}\mathcal{E} = \sum_{i=1}^{N}(\sigma(\mathbf{x}_i;\mathbf{w}_k,b_k) - \eta_{ik})[\mathbf{x}_i^T\ 1]^T \quad (18)$$

$$\nabla_{\mathbf{w}_k,b_k}\nabla_{\mathbf{w}_t,b_t}\mathcal{E} = \sum_{i=1}^{N}\sigma(\mathbf{x}_i;\mathbf{w}_k,b_k)(\delta_K(k,t) - \sigma(\mathbf{x}_i;\mathbf{w}_t,b_t))[\mathbf{x}_i^T\ 1]^T[\mathbf{x}_i^T\ 1] \quad (19)$$

where $\delta_K(.)$ is the Kronecker delta function. The final EM equations are summarized below:

- **Initialization**:

$$[\mathbf{w}_k^T \quad b_k]^T = [0,\ldots,0]^T \quad \forall k \in \{1,...,K\} \quad (20)$$

$$(\pi_m,\mu_m,\Sigma_m)_{m=1}^{M} = \texttt{kmeans}([\mathbf{x}_1\ldots\mathbf{x}_N],M) \quad (21)$$

$$\mathbf{A}_m^j(k_1,k) = \frac{\sum_{i=1}^{N}\delta_K(y_i^j = k_1, y_{i,PLU} = k|z_i = m)}{\sum_{i=1}^{N}\delta_K(y_{i,PLU} = k|z_i = m)} \quad \forall j \in \{1,...R\} \quad (22)$$

$$m \in \{1,...,M\}, k_1 \text{ and } k \in \{1,...,K\}$$

where $\texttt{kmeans}(\mathbf{X}, M)$ is a function that performs K-means clustering over data matrix $\mathbf{X}$ using M clusters, and returns the cluster weights, means and covariance matrices computed from instances in $\mathbf{X}$. $y_{i,PLU}$ is the label obtained by fusing $y_i^1, \ldots, y_i^R$ by simple plurality. The assignment of the i^{th} training instance is done to the closest cluster centroid generated by K-means. Thus, for the purpose of initializing the reliability matrices, the simple plurality label is considered as a proxy for the true label.

- **E-step**:

$$\zeta_{ikm} \propto \pi_m \mathcal{N}(\mathbf{x}_i; \mu_m, \Sigma_m)\sigma(\mathbf{x}_i; \mathbf{w}_k, b_k) \prod_{j=1}^{R} \mathbf{A}_m^j(k_i^j, k) \quad \forall i \in \{1, ...M\}, m \in \{1, ..., M\}$$

$$k \in \{1, ..., K\} \tag{23}$$

$$\gamma_{im} = \sum_{k=1}^{K} \zeta_{ikm} \ , \ \eta_{ik} = \sum_{m=1}^{M} \zeta_{ikm} \quad \forall i \in \{1, ..., N\}, m \in \{1, ..., M\}, k \in \{1, ..., K\} \tag{24}$$

- **M-step**:

$$\pi_m = \frac{\sum_{i=1}^{N} \gamma_{im}}{N} \quad \forall m \in \{1, ..., M\} \tag{25}$$

$$\mu_m = \frac{\sum_{i=1}^{N} \gamma_{im} \mathbf{x}_i}{\sum_{i=1}^{N} \gamma_{im}} \quad \forall m \in \{1, ..., M\} \tag{26}$$

$$\Sigma_m = \frac{\sum_{i=1}^{N} \gamma_{im} (\mathbf{x}_i - \mu_m)(\mathbf{x}_i - \mu_m)^T}{\sum_{i=1}^{N} \gamma_{im}} \quad \forall m \in \{1, ..., M\} \tag{27}$$

$$\mathbf{A}_m^j(k_1, k) = \frac{\sum_{i=1}^{N} \zeta_{ikm} y_{ik_1}^j}{\sum_{i=1}^{N} \zeta_{ikm}} \quad \forall j \in \{1, ...R\}, m \in \{1, ..., M\}$$

$$k_1 \text{ and } k \in \{1, ..., K\} \tag{28}$$

$$(\mathbf{w}_k, b_k)_{k=1}^{K} = \texttt{train-soft-maxent}([\mathbf{x}_1 \ldots \mathbf{x}_N], ([\eta_{i1} \ldots \eta_{iK}])_{i=1}^{N}) \tag{29}$$

$\texttt{train-soft-maxent}(\mathbf{X}, \mathbf{L})$ denotes a function to train a K-class MaxEnt classifier using features in the data matrix $\mathbf{X}$ and soft labels in the $N \times K$ matrix $\mathbf{L}$. Each row of $\mathbf{L}$ contains a probability distribution over the K class labels.

- **Convergence condition**: Terminate the algorithm when the relative change in log-likelihood of the observed data (Eq. 10) is within a specified threshold $\epsilon > 0$, i.e.,

$$1 - \frac{\log P(\mathcal{D}_{obs}|\Theta^{curr})}{\log P(\mathcal{D}_{obs}|\Theta^{prev})} \leq \epsilon \tag{30}$$

Let us consider the variables computed in the E-step. ζ_{ikm} is the probability of the m^{th} mixture component and k^{th} hidden label occurring given the observed data and current parameter values. This can be thought of as a joint soft count of the number of occurrences of $z = m$ and $y = k$. Similarly, γ_{im} and η_{ik} are the individual soft counts of the occurrences of m^{th} mixture component and k^{th} class. It should be noted that while γ_{im} has the same meaning as in EM-based training of a GMM, its expression is different. This is because of the links between observed expert labels y^j and z in Fig. 5, which are absent in the Bayesian network of a simple GMM.

The parameter update equations in the M-step are also intuitively meaningful. The parameters of the GMM are updated as in the case of a simple GMM, except that the soft weights γ_{im} are defined differently. $\mathbf{A}_m^j(k_1, k)$ is equal to a convex combination of the k_1^{th} entry of the 1-in-K encoding of the labels from the j^{th} expert over the database. Put differently, it is proportional to the sum of soft counts ζ_{ikm} over those instances where the j^{th} expert assigned label k_1.

It must be noted that the EM algorithm can get stuck in local maxima of the log-likelihood function. To combat this problem, we allow early stopping of the EM iterations based on the model's accuracy on a development corpus. The next subsection presents a Bayesian version of the model and the associated MAP EM algorithm.

B. A Bayesian Version of the Proposed Model

While the proposed model can account for the data-dependent behavior of experts, it still involves a large number of parameters. This is in spite of the fact that we have constrained the reliability matrices to be the same over each mixture component for a given expert. As shown in Table I, simple plurality is parameter-free and thus does not involve a training stage. The models in [4], [5] and [2] differ by the presence of a K-class classifier in the latter. The proposed model additionally involves $M - 1 + (D + D^2)M$ GMM parameters and M reliability matrices for each expert instead of just 1. As an aside, in terms of computational complexity, simple plurality is roughly $O(1)$, the method by Smyth et al. is $O(RK^2)$, the one by Raykar et al. is $O(RK^2 + KD)$, while the proposed model further scales that complexity by $O(M)$. Thus, training of the proposed model is roughly M-times slower than the one by Raykar et al.

The difference in number of parameters in the proposed model and the model in [2] is $M - 1 + 2DM + (K^2 - K)R(M - 1)$, assuming diagonal covariance matrices for each Gaussian in the GMM. This number is quadratic in K and linear in D, M and R. Thus we expect the

Model	Number of parameters
Simple plurality	0
Smyth et al. [5]	$K + (K^2 - K)R$
Raykar et al. [2]	$(D+1)K + (K^2 - K)R$
Proposed model	$M - 1 + (D + D^2)M + (D+1)K + (K^2 - K)MR$

TABLE I

NUMBER OF PARAMETERS FOR SIMPLE PLURALITY, MODELS PRESENTED IN [4], [5],[2] AND THE PROPOSED MODEL. SIMPLE PLURALITY AND THE MODEL IN [4], [5] DO NOT INVOLVE A K-CLASS CLASSIFIER.

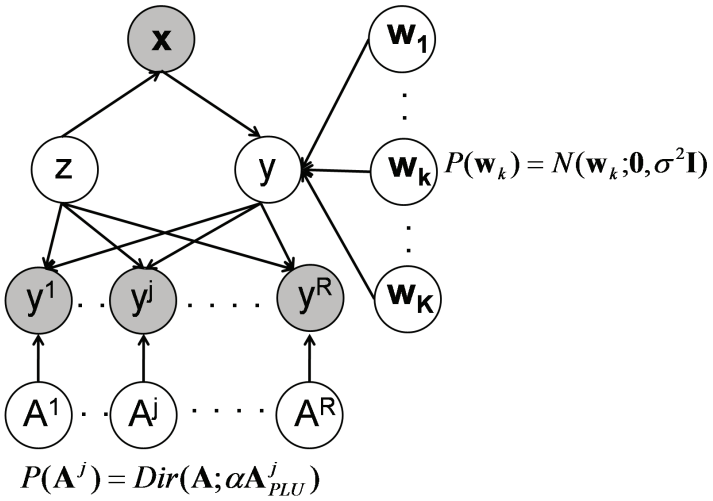

Fig. 6. Bayesian version of the proposed model. All unmentioned PDFs are same as in Fig. 5. Each MaxEnt weight vector $\mathbf{w}_k$ is assumed to be drawn from $\mathcal{N}(\mathbf{w}_k; \mathbf{0}, \sigma^2\mathbf{I})$. $\mathbf{A}^j_{PLU}$ denotes the global reliability matrix of the j^{th} expert computed using the simple plurality labels as a proxy for the true labels. k^{th} column of the reliability matrix $\mathbf{A}^j$ is assumed to be drawn independently from a Dirichlet distribution with parameter vector as α times the k^{th} column of $\mathbf{A}^j_{PLU}$. This is denoted by $P(\mathbf{A}^j) = Dir(\mathbf{A}; \alpha\mathbf{A}^j_{PLU})$.

proposed model to severely overfit the data as all parameters (particularly the number of classes) are increased. One approach to deal with this overfitting is to impose priors on the parameters themselves. We consider two sets of priors – the first one on the MaxEnt parameters (excluding the bias term), and the second one on the expert reliability matrices. This leads us to the Bayesian version of the proposed model as shown in Fig. 6.

We assume that each of the K coefficient vectors in the MaxEnt model (excluding the biases b_k) are generated from a zero mean Gaussian distribution with covariance matrix $\sigma^2\mathbf{I}$. This effectively leads to an L_2 regularization in the MaxEnt objective function. Let $\mathbf{A}^j_{PLU}$ denote the global reliability matrix of the j^{th} expert computed using the simple plurality labels as a proxy for the reference ones. Column k of $\mathbf{A}^j$ is generated from a Dirichlet distribution with a parameter vector which is α times the k^{th} column of $\mathbf{A}^j_{PLU}$. One could assume the variance of the prior distribution of each $\mathbf{w}_k$ to be different. Similarly, each entry of the Dirichlet parameter

vectors could have been tuned independently. But this will result in too many hyperparameters to tune. This model has just two additional hyperparameters to tune – σ and α. Estimation of parameters of this model can be performed using maximum aposteriori (MAP) EM, which requires the computation of the aposteriori PDF of the parameters given the complete data:

$$\log P(\Theta|\mathcal{D}_{obs}, \mathcal{D}_{unobs}) = \log P(\mathcal{D}_{obs}, \mathcal{D}_{unobs}|\Theta) + \log P(\Theta) - \log P(\mathcal{D}_{obs}, \mathcal{D}_{unobs}) \tag{31}$$

The last term is independent of Θ and can be ignored from the optimization. The first term is same as in case of the ML EM and the second term is the prior imposed on the parameters. In the case of the Bayesian network in Fig. 6, this prior can be written as:

$$\log P(\Theta) = \sum_{k=1}^{K} \log \mathcal{N}(\mathbf{w}_k; \mathbf{0}, \sigma^2\mathbf{I}) + \sum_{j=1}^{R} \log Dir(\mathbf{A}^j; \alpha\mathbf{A}^j_{PLU}) \tag{32}$$

The E-step in MAP EM computes the expectation of the PDF $\log P(\Theta|\mathcal{D}_{obs}, \mathcal{D}_{unobs})$ with respect to the posterior PDF of the hidden variables given the observed variables and current estimates of parameters. The M-step maximizes this expectation with respect to the parameters. The final EM equations turn out to be exactly the same as in case of the ML EM algorithm presented earlier, but with the following changes:

1) The estimation equations for the reliability matrices become:

$$\mathbf{A}^j_m(k_1, k) = \frac{\sum_{i=1}^{N} \zeta_{ikm} y^j_{ik_1} + \boldsymbol{\alpha}^j(k_1, k) - 1}{\sum_{i=1}^{N} \zeta_{ikm} + \sum_{k_1=1}^{K} \boldsymbol{\alpha}^j(k_1, k) - K} \quad \forall j \in \{1, ...R\}, m \in \{1, ..., M\}$$
$$k_1 \text{ and } k \in \{1, ..., K\} \tag{33}$$

   where $\boldsymbol{\alpha}^j(k_1, k)$ is the k_1^{th} entry of the k^{th} column of $\alpha\mathbf{A}^j_{PLU}$. We see that the Dirichlet parameters have the effect of increasing the total soft count in the numerator of the above equation. Thus if $\boldsymbol{\alpha}^j(k, k) > \boldsymbol{\alpha}^j(k_1, k) \forall k_1 \neq k$, i.e., the prior count of an expert assigning the correct label is greater than the count of assigning an incorrect label, then $\mathbf{A}^j_m(k, k)$ will be given more additive bias as compared to $\mathbf{A}^j_m(k_1, k)$.

2) The objective function of the soft MaxEnt classifier will now contain an L_2 penalty term, $-\lambda \sum_{k=1}^{K} ||\mathbf{w}_k||^2$ where $\lambda = \frac{1}{2\sigma^2}$. Hence weight vectors with large L_2 norms will be penalized more in the optimization.

The hyper-parameters λ (or σ) and α can be tuned based on classification performance on a development set. Once the parameters of the model have been estimated using either the ML

or MAP criterion, we can perform inference of the hidden label given the feature vector $\mathbf{x}$ and labels from multiple experts as shown in the next subsection.

C. Inference of the Hidden Reference Label

Given a feature vector $\mathbf{x}$ and associated noisy labels from R experts $(y^1, ..., y^R)$, the MAP estimate of the true hidden label y can be found as follows:

$$\begin{aligned}\hat{y}_{MAP} &= \arg\max_k P(y=k|\mathbf{x}, y^1, ..., y^R) \\ &= \arg\max_k \sum_{m=1}^{M} \left\{ \pi_m \mathcal{N}(\mathbf{x}; \mu_m, \mathbf{\Sigma}_m) \exp(\mathbf{w}_k^T \mathbf{x} + b_k) \prod_{j=1}^{R} \prod_{k_1=1}^{K} \mathbf{A}_m^j(k_1, k)^{y_{k_1}^j} \right\} \end{aligned} \tag{34}$$

If the reliability matrices are independent of the mixture component index m (i.e., $\mathbf{A}_m^j(k_1, k) = \mathbf{A}^j(k_1, k) \quad \forall m \in \{1, ..., M\}$), then the decision rule in Eq. 34 reduces to the one in Eq. 9. This implies that the decision rules of the proposed model and the one by Raykar et al. [2] are equivalent if each expert has a single reliability matrix for the entire feature space. We can also conclude that the above decision rule reduces to simple plurality if the sufficient conditions of Prop. 1 are additionaly satisfied.

One could also perform MAP inference of the hidden true label given just the feature vector $\mathbf{x}$ without the noisy labels, corresponding to the practical situation where the experts have not labeled instances in the test set. It can be easily shown that this inference is the same as using the MaxEnt classifier to classify the input feature vector. The next section compares the various models first on simulated data and then on real databases from the UCI repository and two speech corpora for emotion classification. The multiple experts refer to machine classifiers in the case of the UCI databases and human labelers for the emotional speech databases.

IV. Experiments and Results

A. Classification Experiments on Synthetic Data

We conducted classification experiments on synthetic data to understand the behavior of the proposed model. The synthetic database was generated by forward sampling of the Bayesian network shown in Fig. 5. The feature space dimension was set as 2 in a binary classification scenario with 3 experts. The feature vectors were assumed to be generated from a GMM with 4 components, with equal weights assigned to each Gaussian. All covariance matrices were set

to $0.01\mathbf{I}$, where $\mathbf{I}$ is the identity matrix in $\mathbb{R}^2$. The mean vector of the m^{th} Gaussian was set to the components of the m^{th} fourth root of unity:

$$\boldsymbol{\mu}_m(1) = \cos\left(2(m-1)\frac{\pi}{2}\right) \qquad \boldsymbol{\mu}_m(2) = \sin\left(2(m-1)\frac{\pi}{2}\right) \tag{35}$$

The logistic regression weight vectors were set to: $\mathbf{w}_1 = [1\ 1]^T$, $\mathbf{w}_2 = [-1\ \ -1]^T$ and $b_1 = b_2 = 0$. Each reliability matrix had a constant diagonal. The diagonal entries for the first expert's reliability matrices were: $\mathbf{A}_1^1(k,k) = 0.6$, $\mathbf{A}_2^1(k,k) = 0.7$, $\mathbf{A}_3^1(k,k) = 0.8$ and $\mathbf{A}_4^1(k,k) = 0.9$ ($\forall k \in \{1,2\}$), corresponding to four equally spaced points in the interval $[0.55, 0.95]$. Let us represent these diagonal entries by the 4-tuple $(0.6, 0.7, 0.8, 0.9)$. The off-diagonal entries were picked to ensure that each column adds to 1. The diagonal entries for the second and third expert were set to the tuples $(0.9, 0.6, 0.7, 0.8)$ and $(0.8, 0.9, 0.6, 0.7)$ respectively, representing circular right-shifts by indices 1 and 2 of the tuple for the first expert.

Since the two classes are linearly separable, it was observed that all algorithms were able to achieve a very high accuracy (close to 100%). Hence, some noise was introduced in the data generation process. The true labels generated by the MaxEnt model were flipped with probability 0.2 before generating the expert labels. 2500 instances were generated for training, while 1250 instances each were used for development and testing. The development set was used for early stopping of the EM iterations. Both α and λ were set to 0. Fig. 7 shows the inference accuracies of the proposed model and the one by Raykar et al.. Simple plurality and the model by Smyth et al. [5] perform equally well at 79.6%, but significantly worse than the model by Raykar et al. [2]. The inference accuracy of the proposed model is better than all the baselines for $M = 4, 5, 6$ (the correct number of Gaussians was $M = 4$). The accuracy becomes erratic for larger values of M indicating over-fitting. This highlights the fact that the choice of the number of mixture components is extremely critical. We will tune it on a development set in further experiments.

Database	No. of classes	No. of instances	No. of features
Magic Gamma Telescope [19]	2	19020	10
Pima Indians Diabetes [20]	2	768	8
Abalone [21]	2, 3[1]	4177	7
Yeast [10]	4	1484	6
Handwritten Pen Digits [9]	10	10992	16

TABLE II

SUMMARY OF DIFFERENT DATABASES FROM THE UCI REPOSITORY USED IN THE EXPERIMENTS.

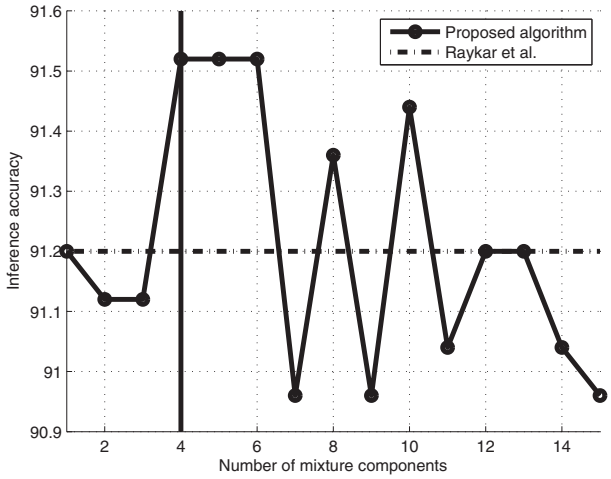

Fig. 7. Inference accuracies of the various models using the synthetic binary classification database. The performance of simple plurality and the algorithm by Smyth et al. was 79.6%. The vertical line corresponds to the true number of Gaussians (4).

B. Classification Experiments on UCI Databases

We next performed classification experiments on 5 chosen databases from the UCI repository [22] for testing the performance of the various models in fusing labels from multiple classifiers. The database details are summarized in Table II. As can be observed, the number of instances, features and classes varies from one database to another. This allows us to test the models on different conditions – binary to multi-class and data rich to data sparse domains. One of the biggest advantages of using these databases is that the reference label is available, making performance evaluation easy.

All databases were split into four sets – training set for the classifiers (30%), training set for the label fusion algorithms (30%), development set for tuning M, α, λ and early stopping of the EM algorithms (20%), and a test set (20%). For ease in setting a range for λ, all features were standardized using the classifier training set. Three standard classifiers from Weka [23] were used as experts – J48 (implementation of the C4.5 decision tree [24]), logistic regression and naive Bayes. Our choice of classifiers was arbitrary and others (like SVM or random forests) could have been selected.

Two sets of experiments were conducted for each of the models – classification using the

[1]The Abalone database had 29 class labels. However the data distribution among these classes is very uneven – 11 classes had less than 20 instances. Hence we converted the problem into binary (age ≤ 9 and ≥ 10) and 3-class (age ≤ 8, $9-10$ and ≥ 11) classification. The class binning was done in a way to ensure that all bins have nearly equal number of samples.

estimated MaxEnt classifier and inference of the true hidden label using the observed data. Inference of the hidden label is done using Eqs. 1, 2, 9 and 34 for simple plurality, the models by Smyth et al. [5] and Raykar et al. [2], and the proposed model, respectively. It must be noted that the first two models do not involve the feature vector $\mathbf{x}$, in contrast to the latter two. Hence inference of y is performed using only the multiple noisy labels in those cases. For computing the classification accuracy, we trained a MaxEnt model separately using the inferred labels for these two models. This was not needed for the model by Raykar et al. [2] and the proposed model since the classifier is already trained as part of the Bayesian network.

The number of Gaussians in the GMM was varied from 1 to $\min(\lfloor N/50 \rfloor, 10)$ (where N is the number of training instances). The upper limit prevents too many Gaussians from being trained for a small training set. It must be noted that the log-likelihood term in the objective function of the L_2 regularized MaxEnt classifier scales as $O(N)$ (where N is the number of training instances), making the penalty term $(-\lambda \sum_{k=1}^{K} ||\mathbf{w}_k||^2)$ negligible in magnitude. Thus, the regularization parameter λ set equal to βN, where β was varied from 0 to 0.1 in steps of 0.005. The Dirichlet parameter α was varied from 0 to 0.2 in steps of 0.02. Larger values of α resulted in excessive smoothing and hence poorer performance.

Classifier/Model	Classification Accuracy	Inference Accuracy
J48	83.88	-
Logistic	79.77	-
Naive Bayes	72.99	-
Simple plurality	79.48 ($\beta = 0$)	81.97
Smyth et al.	79.49 ($\beta = 0$)	81.97
Raykar et al.	79.72 ($\beta = 0, \alpha = 0.2$)	81.71 ($\beta = 0, \alpha = 0.08$)
GVLC model	**80.04** ($\beta = 0, \alpha = 0.14, M = 10$)	81.71 ($\beta = 0, \alpha = 0.08, M = 1$)
GVLC model (oracle)	**80.12** ($\beta = 0, \alpha = 0, M = 2$)	81.87 ($\beta = 0, \alpha = 0, M = 3$)

TABLE III

CLASSIFICATION AND INFERENCE ACCURACIES FOR THE MAGIC GAMMA TELESCOPE DATABASE.

Tables III-VIII show the accuracies for the different databases on the test set. The first three rows represent accuracies of the three classifiers. The oracle accuracy of the globally-varying locally-constant (GVLC) model is obtained by tuning all the hyperparameters on the test set and gives an upper bound of the model's performance. It is expected that with lesser mismatch between development and test set, and finer parameter tuning, one can come very close to this bound. Values in bold represent a statistically significant improvement in performance over

Classifier/Model	Classification Accuracy	Inference Accuracy
J48	72.56	-
Logistic	79.27	-
Naive Bayes	76.83	-
Simple plurality	76.83 ($\beta = 0.005$)	79.27
Smyth et al.	76.83 ($\beta = 0.005$)	79.27
Raykar et al.	77.44 ($\beta = 0.015, \alpha = 0$)	78.66 ($\beta = 0.035, \alpha = 0.06$)
GVLC model	**78.05** ($\beta = 0.005, \alpha = 0.08, M = 2$)	79.27 ($\beta = 0.1, \alpha = 0.12, M = 3$)
GVLC model (oracle)	**78.66** ($\beta = 0.005, \alpha = 0.04, M = 3$)	**80.49** ($\beta = 0.04, \alpha = 0.16, M = 3$)

TABLE IV

CLASSIFICATION AND INFERENCE ACCURACIES FOR THE PIMA INDIANS DATABASE.

Classifier/Model	Classification Accuracy	Inference Accuracy
J48	77.09	-
Logistic	78.79	-
Naive Bayes	73.45	-
Simple plurality	74.91 ($\beta = 0.005$)	78.42
Smyth et al.	74.91 ($\beta = 0.005$)	78.42
Raykar et al.	**76.12** ($\beta = 0.005, \alpha = 0$)	78.42 ($\beta = 0.02, \alpha = 0.12$)
GVLC model	**76.36** ($\beta = 0.005, \alpha = 0, M = 4$)	**79.15** ($\beta = 0.005, \alpha = 0.04, M = 4$)
GVLC model (oracle)	**76.73** ($\beta = 0.005, \alpha = 0, M = 3$)	**79.88** ($\beta = 0, \alpha = 0, M = 9$)

TABLE V

CLASSIFICATION AND INFERENCE ACCURACIES FOR THE ABALONE DATABASE (BINARY).

Classifier/Model	Classification Accuracy	Inference Accuracy
J48	61.18	-
Logistic	66.59	-
Naive Bayes	57.57	-
Simple plurality	62.38 ($\beta = 0.005$)	64.54
Smyth et al.	62.50 ($\beta = 0.005$)	64.54
Raykar et al.	62.26 ($\beta = 0.005, \alpha = 0.20$)	64.66 ($\beta = 0.005, \alpha = 0.04$)
GVLC model	**63.94** ($\beta = 0.005, \alpha = 0.12, M = 6$)	65.02 ($\beta = 0.025, \alpha = 0.20, M = 9$)
GVLC model (oracle)	**64.90** ($\beta = 0.005, \alpha = 0.16, M = 9$)	**65.99** ($\beta = 0.010, \alpha = 0.16, M = 4$)

TABLE VI

CLASSIFICATION AND INFERENCE ACCURACIES FOR THE ABALONE DATABASE (3-CLASS).

Classifier/Model	Classification Accuracy	Inference Accuracy
J48	53.29	-
Logistic	55.71	-
Naive Bayes	57.09	-
Simple plurality	56.06 ($\beta = 0.005$)	55.02
Smyth et al.	56.06 ($\beta = 0.005$)	55.71
Raykar et al.	56.06 ($\beta = 0.005, \alpha = 0$)	55.02 ($\beta = 0.005, \alpha = 0$)
GVLC model	57.09 ($\beta = 0.010, \alpha = 0.10, M = 4$)	**57.09** ($\beta = 0.075, \alpha = 0.08, M = 3$)
GVLC model (oracle)	**57.99** ($\beta = 0.010, \alpha = 0.12, M = 3$)	**58.48** ($\beta = 0.005, \alpha = 0, M = 4$)

TABLE VII

CLASSIFICATION AND INFERENCE ACCURACIES FOR THE YEAST DATABASE.

Classifier/Model	Classification Accuracy	Inference Accuracy
J48	93.17	-
Logistic	95.03	-
Naive Bayes	85.78	-
Simple plurality	90.43 ($\beta = 0$)	95.03
Smyth et al.	90.71 ($\beta = 0$)	94.89
Raykar et al.	**90.85** ($\beta = 0, \alpha = 0$)	94.80 ($\beta = 0, \alpha = 0.12$)
GVLC model	**92.15** ($\beta = 0, \alpha = 0.12, M = 8$)	**95.45** ($\beta = 0, \alpha = 0.12, M = 4$)
GVLC model (oracle)	**92.15** ($\beta = 0, \alpha = 0.12, M = 8$)	**95.91** ($\beta = 0, \alpha = 0.18, M = 3$)

TABLE VIII

CLASSIFICATION AND INFERENCE ACCURACIES FOR THE HANDWRITTEN PEN DIGITS DATABASE.

simple plurality at the 5% significance level using the exact one-sided binomial test[2]. As can be observed, the GVLC model gives the highest classification accuracy for 6 databases and the highest inference accuracy for 4 databases (except the Magic Gamma Telescope and Pima Indians database). The improvement is statistically significant with respect to simple plurality for 5 and 3 databases for classification and inference respectively. In contrast, the algorithm by Raykar et al. gives a statistically significant improvement in only 2 databases for classification and none for inference. The oracle GVLC model gives a statistically significant improvement for 6 and 5 databases in classification and inference respectively, indicating that careful tuning of the hyperparameters is crucial.

C. Emotion Classification from Speech

We consider here the problem of human emotion recognition from speech. As mentioned earlier, even though human emotion expressions span a continuum, they are often quantized into categories such as $\{$angry, happy, sad, neutral$\}$. Labeling human speech for emotions is a difficult task and multiple human evaluators are typically used. In this work, we use two emotional speech databases for our experiments. The first database [25] (called the EMA database) has 3 trained actors reading 10 sentences 5 times each portraying the four aforementioned emotional states. This results in 150 audio clips per emotional class. All the clips were then labeled by 4 human evaluators who assigned a class label to each clip. The emotion which the actor was asked to synthesize was assumed to be the reference label. We extracted the root mean squared energy (RMSE) and 12 Mel-Frequency Cepstral Coefficients (MFCCs) over 20 ms frames with 10 ms

[2]We did not use a Chi-squared or McNemar's test since they require distributional assumptions. Consider a performance comparison between algorithms 1 and 2. Let $n_{i,j}$ be the number of times algorithm i is correct and j is incorrect, where $i, j \in \{1, 2\}$. The exact binomial test checks the null hypothesis that n_{12} and n_{21} are counts of head and tail respectively from a fair coin. The alternate hypothesis is that the coin has $P(\text{head}) > 0.5$. The p-value of this test can be computed from the binomial CDF with parameter $P(\text{head}) = 0.5$. We consider $p \leq 0.05$ as a significant result.

shift using the OpenSMILE toolkit [26]. The component-wise mean of this feature vector was computed over each utterance resulting in a 13-dimensional utterance-level feature vector. The data was randomly split into three sets for training (40%), testing (30%) and development (30%). Similar to the procedure adopted for the UCI databases, we standardized the features using mean and variance computed from the training set. Table IX shows the emotion classification and inference accuracy for the various models. The proposed model performs better than the three baseline models both in classification and inference of the true emotion label. In addition, it is the only model which achieves statistically significant improvements over simple plurality.

Model	Classification Accuracy	Inference Accuracy
Annotator 1	93.37	-
Annotator 2	90.36	-
Annotator 3	98.19	-
Annotator 4	77.71	-
Simple plurality	81.93 ($\beta = 0.025$)	98.80
Smyth et al.	82.53 ($\beta = 0.025$)	96.99
Raykar et al.	82.53 ($\beta = 0.025, \alpha = 0$)	98.80 ($\beta = 0, \alpha = 0$)
GVLC model	**84.94** ($\beta = 0.015, \alpha = 0.04, M = 1$)	**99.40** ($\beta = 0, \alpha = 0, M = 2$)
GVLC model (oracle)	**86.14** ($\beta = 0, \alpha = 0, M = 4$)	**99.40** ($\beta = 0, \alpha = 0, M = 2$)

TABLE IX

EMOTION CLASSIFICATION AND INFERENCE ACCURACIES FOR THE EMA DATABASE.

Apart from modeling categorical representations, research in the emotional speech analysis community has also focussed on other dimensional representations. The most popular of these are activation and valence [27]. Valence is a bipolar rating of the pleasantness of speech. Activation on the other hand, denotes the excitation in speech. For example, the emotional class angry is expected to have negative valence and positive activation. To further test the performance of various models, we conducted valence and activation classification experiments on the same database as above. Following the convention, {angry, happy} were assigned high and {sad, neutral} were assigned low activation. For valence, {angry, sad} were labeled as negative while {happy, neutral} were labeled as positive. Tables X-XI show the classification and inference accuracies for these two cases. While the GVLC model gives a statistically significant improvement over simple plurality, it performs as good as the baseline models. This is understandable in the case of inference, since the accuracy is already extremely high. We attempt to explain this observation in Subsection IV-D. Before that, we present results on the SEMAINE database.

The SEMAINE database [28] is a large multimodal, audio-visual database, collected as part

Model	Classification Accuracy	Inference Accuracy
Annotator 1	95.36	-
Annotator 2	94.70	-
Annotator 3	98.01	-
Annotator 4	84.77	-
Simple plurality	82.78 ($\beta = 0$)	98.01
Smyth et al.	**84.77** ($\beta = 0$)	**98.68**
Raykar et al.	**84.11** ($\beta = 0, \alpha = 0$)	**98.68** ($\beta = 0, \alpha = 0.2$)
GVLC model	**84.77** ($\beta = 0, \alpha = 0.02, M = 3$)	**98.68** ($\beta = 0, \alpha = 0.2, M = 1$)
GVLC model (oracle)	**86.09** ($\beta = 0.005, \alpha = 0.02, M = 3$)	**98.68** ($\beta = 0, \alpha = 0, M = 1$)

TABLE X

VALENCE CLASSIFICATION AND INFERENCE ACCURACIES FOR THE EMA DATABASE.

Model	Classification Accuracy	Inference Accuracy
Annotator 1	99.36	-
Annotator 2	96.79	-
Annotator 3	99.36	-
Annotator 4	83.33	-
Simple plurality	92.99 ($\beta = 0.005$)	96.18
Smyth et al.	**95.54** ($\beta = 0.005$)	**99.36**
Raykar et al.	**95.54** ($\beta = 0.005, \alpha = 0$)	**100.00** ($\beta = 0.005, \alpha = 0.2$)
GVLC model	**95.54** ($\beta = 0.005, \alpha = 0, M = 1$)	**99.36** ($\beta = 0.005, \alpha = 0, M = 1$)
GVLC model (oracle)	**99.36** ($\beta = 0.005, \alpha = 0.20, M = 1$)	**100.00** ($\beta = 0, \alpha = 0, M = 1$)

TABLE XI

ACTIVATION CLASSIFICATION AND INFERENCE ACCURACIES FOR THE EMA DATABASE.

of a research effort to build Sensitive Artificial Listener (SAL) agents. These agents should be able to interact with a human in a sustained, emotionally colored conversation. All interactions involve two persons, a human user and an operator (who can be either a machine or a human simulating a machine agent). The operator has four personalities – Spike (angry), Poppy (happy), Obadiah (sad) and Prudence (sensible/neutral). The operator's task is to induce his/her own personality into the user as the conversation goes along. There are a total of 94 sessions in the SEMAINE database, where each session includes audio and video recordings of the interaction. Each session is rated by multiple human evaluators for various emotional dimensions (such as valence, activation, power and intensity), and characteristics of the interaction (such as breakdown of engagement, social concealment etc). For the purpose of this paper, we pick three emotional dimensions – valence, activation and intensity, for comparing various algorithms. Intensity captures how far the speaker is from a state of cool rationality, irrespective of the direction. Valence and activation have the same meanings as explained earlier. Since the emotion of the character being played by the human operator is clearly defined, we decided to use just its audio. Furthermore, only sessions 19, 20, 21 and 22 (Obadiah, Spike, Poppy and Prudence respectively) contained ratings from the same set of 3 human evaluators (evaluators R1, R2 and

R3). Hence, we only used these sessions for our experiments. For each session, the ratings from an annotator are recorded as a time series and are available every 20 ms. As a pre-processing step, we segmented the operator's audio into sentences using the time-aligned text transcriptions available in the database. Next, RMS energy and 12 MFCC features were extracted over 20 ms frames with 10 ms shift from each sentence using the OpenSMILE toolkit. These 13-dimensional feature vectors and evaluator ratings were averaged over 5 contiguous frames, since sentence-level averaging would have resulted in much fewer instances. We ended up with 4452 instances.

Since the dimensional ratings were continuous, they had to be quantized appropriately before various models could be trained. It was observed that the valence, activation and intensity ratings are well-represented by 3, 2 and 2 clusters respectively. This was corroborated by observing the ratings for the different operator personalities. In terms of activation, Prudence clearly falls in the low activation category along with Obadiah. However, its valence ratings are neither extremely positive nor extremely negative. Thus, we quantized the valence and activation ratings into 3 and 2 levels respectively using K-means for each evaluator independently. Intensity was represented by 2 clusters since Prudence usually had a low rating, while the other 3 personalities had high ratings. The reference labels for valence classification were obtained by assigning label 1 to Obadiah and Spike, 2 to Prudence and 3 to Poppy. Similarly, the reference activation labels were obtained by mapping Obadiah and Prudence to 1, and Spike and Poppy to 2. For intensity, we assigned 1 to Prudence and 2 to the remaining personalities. This reference label assignment was corroborated by the histograms of the average evaluator continuous ratings. The 4452 instances were split into a training (40%), test (30%) and development set (30%). Tables XII-XIV show the valence, activation and intensity classification accuracies of various algorithms. The GVLC model gives statistically significant improvement over simple plurality for valence inference and activation/intensity classification.

D. An Insight into GVLC Model's Benefit

The proposed GVLC model gives statistically significant improvement over simple plurality for 10 and 7 out of 12 test cases (for classification and inference respectively). This is appreciably better than the performance of the next best algorithm (one by Raykar et al.), which achieves a statistically significant improvement in only 4 cases for classification and 2 for inference. It is interesting to note that the benefit obtained by the proposed data-dependent model over the

Model	Classification Accuracy	Inference Accuracy
Annotator 1	96.71	-
Annotator 2	92.23	-
Annotator 3	92.31	-
Simple plurality	52.95 ($\beta = 0.005$)	96.56
Smyth et al.	52.85 ($\beta = 0.005$)	96.56
Raykar et al.	53.02 ($\beta = 0.005, \alpha = 0.04$)	96.56 ($\beta = 0, \alpha = 0$)
GVLC model	53.25 ($\beta = 0.005, \alpha = 0.08, M = 7$)	**96.79** ($\beta = 0, \alpha = 0.06, M = 9$)
GVLC model (oracle)	**53.32** ($\beta = 0.005, \alpha = 0, M = 7$)	**96.79** ($\beta = 0, \alpha = 0.06, M = 9$)

TABLE XII

VALENCE CLASSIFICATION AND INFERENCE ACCURACIES FOR THE SEMAINE DATABASE.

Model	Classification Accuracy	Inference Accuracy
Annotator 1	98.29	-
Annotator 2	63.05	-
Annotator 3	94.72	-
Simple plurality	73.16 ($\beta = 0.015$)	97.25
Smyth et al.	73.16 ($\beta = 0.015$)	97.25
Raykar et al.	73.31 ($\beta = 0.015, \alpha = 0.06$)	96.88 ($\beta = 0.005, \alpha = 0.08$)
GVLC model	**73.75** ($\beta = 0.015, \alpha = 0.02, M = 5$)	97.03 ($\beta = 0, \alpha = 0.04, M = 9$)
GVLC model (oracle)	**73.75** ($\beta = 0.015, \alpha = 0, M = 5$)	97.25 ($\beta = 0, \alpha = 0, M = 7$)

TABLE XIII

ACTIVATION CLASSIFICATION AND INFERENCE ACCURACIES FOR THE SEMAINE DATABASE.

data-independent one by Raykar et al. is variable across databases. An insight into this variation can be gained by recalling the essential difference between the two models. While the model by Raykar et al. assumes each expert's reliability matrix to be constant over the entire feature space, the GVLC model makes this assumption only over clusters of homogeneous instances. Thus, it is natural to expect that the GVLC model gives greater performance benefit for databases with greater variation of expert reliability over the feature space. To check this intuition, we define a measure of the variation of an expert's reliability given a database.

Consider a feature space consisting of M clusters of data instances derived from some clustering algorithm (we used K-means). Assuming we have the true reference labels available,

Model	Classification Accuracy	Inference Accuracy
Annotator 1	97.15	-
Annotator 2	90.77	-
Annotator 3	98.87	-
Simple plurality	63.66 ($\beta = 0.02$)	97.30
Smyth et al.	63.66 ($\beta = 0.02$)	97.30
Raykar et al.	63.81 ($\beta = 0.025, \alpha = 0$)	97.30 ($\beta = 0, \alpha = 0$)
GVLC model	**64.34** ($\beta = 0.06, \alpha = 0, M = 9$)	97.30 ($\beta = 0, \alpha = 0, M = 1$)
GVLC model (oracle)	**64.64** ($\beta = 0.065, \alpha = 0, M = 1$)	97.30 ($\beta = 0, \alpha = 0, M = 1$)

TABLE XIV

INTENSITY CLASSIFICATION AND INFERENCE ACCURACIES FOR THE SEMAINE DATABASE.

we can estimate the global $K \times K$ reliability matrix of the j^{th} expert – $\mathbf{A}^j_{glob}$, where $\mathbf{A}^j_{glob}(k_1, k) = P(y^j = k_1 | y = k)$. Similarly, we can estimate a local reliability matrix for all instances belonging to the m^{th} cluster – $\mathbf{A}^j_{loc,m}$. The reliability variation of expert j given cluster m and reference label k can be defined as:

$$\mathcal{RV}^j(k, m) = \Big(JSD(\mathbf{A}^j_{glob}(:, k), \mathbf{A}^j_{loc,m}(:, k)) \Big)^{1/2} \tag{36}$$

where $JSD(p, q)$ denotes the Jensen-Shannon divergence [29] between probability mass functions p and q, and $\mathbf{X}(:, k)$ denotes the k^{th} column of $\mathbf{X}$. Taking square root makes the Jensen-Shannon divergence a metric. Now, the expected reliability variation of expert j can be computed as:

$$\mathbb{E}\{\mathcal{RV}^j\} = \sum_{m=1}^{M} \sum_{k=1}^{K} \mathcal{RV}^j(k, m) P(y = k, z = m) \tag{37}$$

Upon averaging $\mathbb{E}\{\mathcal{RV}^j\}$ over all experts for a given database, we get a metric representing the average reliability variation of the experts from their global reliabilities. We next compute the correlation coefficient between this average metric and the relative inference performance improvement obtained by the proposed model with respect to the model by Raykar et al. [2] over all 12 test cases. The oracle performance was used since it guards against any noise introduced due to hyper-parameter mismatch. The correlation coefficient was found to be 0.74 (significant at the 5% level). This indicates that the benefit of the proposed model is more when expert reliability is highly variable over the feature space. In case the reliability is nearly constant, overfitting due to increase in the number of parameters nullifies any gain obtained by modeling the reliability variation.

V. Conclusions and Future Work Directions

This paper presented a model for capturing the data-dependent behavior of experts. It is based on the observation that experts (whether machine classifiers or human evaluators) do not have equal reliability for all instances processed. Rather, their reliability varies from one region of the feature space to another. In this work, this reliability is assumed to be constant over clusters of instances in the feature space and can be modeled by a Gaussian mixture model. This enables us to model experts as having a reliability matrix for each Gaussian in the mixture. The hidden reference label is assumed to be generated from the feature vector using a MaxEnt model. This

hidden label is then distorted by each expert using the reliability matrix corresponding to the Gaussian which generated the feature vector. All the parameters of this model are learned in the ML sense using the EM algorithm. A Bayesian version of this model is also proposed, where the MaxEnt classifier coefficients are L_2-regularized and the expert reliability matrix entries are generated from a Dirichlet distribution. Experiments on simulated data, a variety of databases from the UCI repository, and two emotion classification databases show improvements in both classification and inference accuracy by using the proposed model.

There are many interesting directions for future work in this domain. First, this paper assumes that all the instances are independent. However many practical problems involve labeling of time series. For example, speaker clustering and diarization involve labeling frames of an audio clip with speaker indices. Labeling of human body motion capture data for special events of interest (e.g., body gestures) is another example. It would be interesting to extend the proposed model to handle multiple labeling of time series. One simple way to do this is to impose a first order Markov chain structure on the temporal evolution of the hidden variable z in Fig. 5, resulting in a Dynamic Bayesian Network (DBN).

Second, in the case of most classifiers as experts, it is easy to generate a complete posterior distribution over the label set for each instance. The problem now modifies to inferring the true posterior distribution (or just the true label) using posterior distributions from multiple noisy experts. Jin and Ghahramani [30] consider a related problem where each training instance is associated with a subset of labels, exactly one of which is correct. A more general version of this problem involves a single posterior distribution over labels associated with each instance. In the case of human experts, it is tough to get a posterior distribution. But a ranked list of labels is much easier to obtain. Thus the problem can be modified to devising a scheme for a combination of ranked lists from multiple human experts. It would be interesting to see if a more structured way of inferring the true hidden label from multiple noisy ranked lists gives benefits over standard voting algorithms like the Borda count and Schulze's method [31], [32].

REFERENCES

[1] E. Mower, A. Metallinou, C-C. Lee, A. Kazemzadeh, C. Busso, S. Lee, and S. Narayanan, "Interpreting ambiguous emotional expressions," in *Proc. ACII*, Sept. 2009.

[2] V. C. Raykar, S. Yu, L. S. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy, "Learning from Crowds," *Journal of Machine Learning Research*, vol. 11, pp. 1297–1322, Mar. 2010.

[3] R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng, "Cheap and fast—but is it good?: Evaluating non-expert annotations for natural language tasks," in *Proc. EMNLP*, 2008, pp. 254–263.

[4] A. P. Dawid and A. M. Skene, "Maximum likelihood estimation of observer error-rates using the EM algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 20–28, 1979.

[5] P. Smyth, U. Fayyad, M. Burl, P. Perona, and P. Baldi, "Inferring ground truth from subjective labeling of Venus images," in *Proc. NIPS*, 1995, pp. 1085–1092.

[6] K. Audhkhasi and S. S. Narayanan, "Data-dependent evaluator modeling and its application to emotional valence classification from speech," in *Proc. Interspeech*, 2010.

[7] A. P. Dempster, N. M. Liard, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of Royal Statistical Society: Series B*, vol. 39, pp. 1–38, 1977.

[8] M. Grimm and K. Kroschel, "Evaluation of natural emotions using self assessment manikins," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2005, pp. 381–385.

[9] F. Alimoglu and E. Alpaydin, "Methods of Combining Multiple Classifiers Based on Different Representations for Pen-based Handwritten Digit Recognition," in *Proc. Fifth Turkish Artificial Intelligence and Artificial Neural Networks Symposium (TAINN 96)*, 1996.

[10] P. Horton and K. Nakai, "A probabilistic classification system for predicting the cellular localization sites of proteins," in *Proc. Fourth International Conference on Intelligent Systems for Molecular Biology*, 1996, pp. 109–115.

[11] J. Whitehill, P. Ruvolo, T. Wu, J. Bergsma, and J. Movellan, "Whose vote should count more: Optimal integration of labels from labelers of unknown expertise," in *Proc. NIPS*, 2009, vol. 22, pp. 2035–2043.

[12] Y. Yan, R. Rosales, G. Fung, M. Schmidt, G. Hermosillo, L. Bogoni, L. Moy, and G. J. Dy, "Modeling annotator expertise: Learning when everyone knows a bit of something," in *Proc. AISTATS*, 2010.

[13] M. Marge, S. Banerjee, and A. I. Rudnicky, "Using the Amazon Mechanical Turk for transcription of spoken language," in *Proc. ICASSP*, 2010.

[14] A. Sorokin and D. Forsyth, "Utility data annotation with Amazon Mechanical Turk," in *Proc. CVPR*, 2008, pp. 1–8.

[15] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR*, 2009.

[16] J. Heer and M. Bostock, "Crowdsourcing graphical perception: using Mechanical Turk to assess visualization design," in *Proc. Intl. Conf. on Human Factors in Computing Systems*, 2010, pp. 203–212.

[17] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, vol. 2, Wiley, New York, 2001.

[18] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[19] D. Heck, J. Knapp, J. Capdevielle, G. Schatz, and T. Thouw, *CORSIKA: A Monte Carlo code to simulate extensive air showers*, vol. 6019, FZKA 6019 Forschungszentrum Karlsruhe, 1998.

[20] J.W. Smith, J. E. Everhart, W. C. Dickson, W. C. Knowler, and R.S. Johannes, "Using the ADAP learning algorithm to forecast the onset of diabetes mellitus," in *Proc. Annual Symposium on Computer Application in Medical Care*, 1988, p. 261.

[21] W.J. Nash, *The Population Biology of Abalone (Haliotis Species) in Tasmania. I. Blacklip Abalone (H Rubra) from the North Coast and the Islands of Bass Strait*, Sea Fisheries Division, Marine Research Laboratories – Taroona, Dept. of Primary Industry and Fisheries, Tasmania, 1978.

[22] A. Frank and A. Asuncion, "UCI machine learning repository," *http://archive.ics.uci.edu/ml*, 2010.

[23] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I.H. Witten, "The WEKA data mining software: an update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[24] J.R. Quinlan, *C4.5: programs for machine learning*, Morgan Kaufmann, 1993.

[25] S. Lee, S. Yildirim, A. Kazemzadeh, and S. Narayanan, "An articulatory study of emotional speech production," in *Proc. Ninth European Conference on Speech Communication and Technology*, 2005.

[26] F. Eyben, M. Wöllmer, and B. Schuller, "OpenSMILE: The Munich versatile and fast open-source audio feature extractor," in *Proc. ACM Multimedia*, 2010, pp. 1459–1462.

[27] R. Kehrein, "The prosody of authentic emotions," in *Proc. Speech Prosody Conference*, 2002, pp. 423–426.

[28] G. McKeown, M. Valstar, R. Cowie, M. Pantic, and M. Schroder, "The SEMAINE database: Annotated multimodal records of emotionally coloured conversations between a person and a limited agent," *IEEE Trans. on Affective Computing*, pp. 1–14, 2011.

[29] J. Lin, "Divergence measures based on the Shannon entropy," *IEEE Trans. on Information Theory*, vol. 37, no. 1, pp. 145–151, 1991.

[30] R. Jin and Z. Ghahramani, "Learning with multiple labels," *Proc. NIPS*, pp. 921–928, 2003.

[31] M. Schulze, "A new monotonic and clone-independent single-winner election method," *Voting matters*, vol. 17, pp. 9–19, 2003.

[32] D. Black, *The theory of committees and elections*, Springer, 1986.

Kartik Audhkhasi received the B.Tech. degree in Electrical Engineering and M.Tech. degree in Information and Communication Technology from Indian Institute of Technology, Delhi in 2008. He is currently pursuing the Ph.D. degree in Electrical Engineering from University of Southern California, Los Angeles. His research interests are in modeling, analysis and design of ensembles of multiple human experts or machine classifiers. He is also interested in crowd-sourcing techniques for speech and language processing. Kartik is the recipient of the Annenberg and IBM PhD fellowships, and best teaching assistant awards of the EE department at USC.

Shrikanth (Shri) Narayanan is the Andrew J. Viterbi Professor of Engineering at the University of Southern California (USC), and holds appointments as Professor of Electrical Engineering, Computer Science, Linguistics and Psychology and as the founding director of the Ming Hsieh Institute. Previously he was with AT&T Labs-Research, Florham Park and AT&T Bell Labs, Murray Hill, New Jersey. He is a Fellow of the Acoustical Society of America, the Institute of Electrical and Electronics Engineers and the American Association for the Advancement of Science (AAAS). His research interests are in signals and systems modeling with an interdisciplinary emphasis on speech, audio, language, multimodal and biomedical problems and applications with direct societal relevance. He has published over 450 papers and has 12 granted U.S. patents.