A Survey on Epistemic (Model) Uncertainty in Supervised Learning: Recent Advances and Applications

Xinlei Zhou[a], Han Liu[a,b], Farhad Pourpanah[b,c], Tieyong Zeng[d], Xizhao Wang[a,b,*]

[a] *College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China*
[b] *The Guangdong Key Laboratory of Intelligent Information Processing, Shenzhen University, 518060, China*
[c] *College of Mathematics and Statistics, Shenzhen University, Shenzhen 518060, China*
[d] *Department of Mathematics, The Chinese University of Hong Kong, Hong Kong*

Abstract

Quantifying the uncertainty of supervised learning models plays an important role in making more reliable predictions. *Epistemic* uncertainty, which usually is due to insufficient knowledge about the model, can be reduced by collecting more data or refining the learning models. Over the last few years, scholars have proposed many epistemic uncertainty handling techniques which can be roughly grouped into two categories, i.e., Bayesian and ensemble. This paper provides a comprehensive review of epistemic uncertainty learning techniques in supervised learning over the last five years. As such, we, first, decompose the epistemic uncertainty into bias and variance terms. Then, a hierarchical categorization of epistemic uncertainty learning techniques along with their representative models is introduced. In addition, several applications such as computer vision (CV) and natural language processing (NLP) are presented, followed by a discussion on research gaps and possible future research directions.

Keywords: Epistemic uncertainty learning, supervised learning, Bayesian approximation, ensemble learning, computer vision, natural language processing

1. Introduction

Supervised learning, as a broad branch of machine learning, refers to the task of learning a mapping function for associating high-dimensional input samples into their corresponding target vectors using labeled data [1, 2, 3, 4]. They have been successfully used for a variety of real-world applications, e.g., medical and fault diagnosis [5, 6, 7, 8], object detection [9, 10], text processing [11, 12, 13], and speech recognition [14], image segmentation [15, 16, 17], image enhancement [18, 19]. Indeed, supervised learning is a process of predicting unknown data based on partial

*Corresponding author
Email addresses: Zhouxnli@gmail.com (Xinlei Zhou), han.liu@szu.edu.cn (Han Liu), farhad@szu.edu.cn,farhad.086@gmail.com (Farhad Pourpanah), zeng@math.cuhk.edu.hk (Tieyong Zeng), xizhaowang@ieee.org (Xizhao Wang)

samples that cannot accurately represent the whole data set distribution. In such an experience-driven process, the model is not provably correct but only hypothetical; therefore uncertain and the same holds for the predictions produced by the model [20]. In addition, the challenge of big data, such as skyrocketed feature dimensions and categories, missing data, unbalanced data distribution and huge solution space, aggravate the uncertainty of the learning process, which seriously affects the performance of the supervised learning algorithms [21]. Moreover, supervised learning approaches are unable to identify in-domain from out-domain samples [22], provide reliable uncertainty approximation [23], and lack expressiveness during inference [24]; therefore, their deployment in high-risk and safety-critical applications remains limited. To alleviate these issues, it is vital to present uncertainty estimate in a way that ignores the uncertain predictions or passes them to experts [25].

In supervised learning, traditional uncertainty assessment is usually based on a single probability distribution. Nowadays, the widely accepted way is quantifying uncertainty separately by distinguishing two different sources, i.e., *aleatoric* uncertainty and *epistemic* uncertainty [20]. Aleatoric (data) uncertainty is a kind of uncertainty that reflects the inherent property of data, like noise. It is usually caused by the irreducible error in the data measurements and observations process, which cannot be reduced even by collecting more data. Kendal and Gal [26] further divided aleatoric uncertainty into *homoscedastic* uncertainty and *heteroscedastic* uncertainty. The former is a value that stays constant for various input samples in the same task and the latter is associated with the differences among input data, for example, some inputs contain more noise than others. In contrast, epistemic (model) uncertainty is referred to a state that model cognition is restricted, which is due to the upper limit of the model fitting ability, the optimizing strategy, the parameters, the lack of knowledge. It can be reduced by gathering more data or refining models.

On the other hand, the generalization error is a standard metric to quantify the effectiveness of decisions made by supervised learning models. Meanwhile, several studies [27, 28, 29] have proved that the generalization error manifests the predictive uncertainty. It simultaneously commits to the theoretical exploration of the quantification and formal expression of their relationship. The generalization error can be decomposed into three terms, i.e., noise, bias, and variance [30, 31].

Suppose, $y_o = f(\mathbf{x}) + \epsilon$ represents the observed value for a given $\mathbf{x} \in \Re^d$, which is corrupted by noise $\epsilon \sim N(0, \sigma_1^2)$. Thus, $y_o \sim N(f(\mathbf{x}), \sigma_1^2)$ where $f(\mathbf{x})$ represents the original target. Besides, $y_p \sim N(\hat{y}, \sigma_2^2)$ denotes the distribution of prediction $\hat{f}(\mathbf{x})$ centered on mean value $\hat{y} = \mathbb{E}\hat{f}(\mathbf{x})$. The noise term (σ_1^2) arises from data and it is irreducible, which can be represented as aleatoric (data) uncertainty. In contrast, the (squared) bias term $([\mathbb{E}\hat{f}(\mathbf{x}) - f(\mathbf{x})]^2)$ reveals the gap between the estimated value and the true value. It reflects the degree of cognitive limitation caused by the setting of model properties such as parameters, strategies, or learning algorithms. While, the variance term $(\mathbb{E}[\hat{f}(\mathbf{x}) - \mathbb{E}\hat{f}(\mathbf{x})]^2)$ is related to the sensitivity of model pertaining to the training samples. Thus, we argue that the bias term together with variance represents

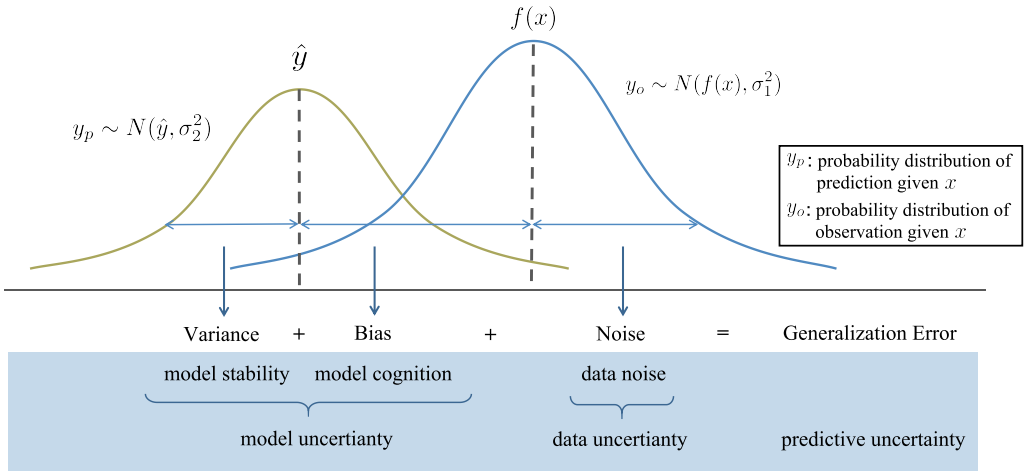

Fig. 1: Decomposition of uncertainty in supervised learning.

the epistemic (model) uncertainty. Fig. 1 shows these three terms and explains the predictive uncertainty from the perspective of the generalization error decomposition. The generalization error of a model can be reduced through the correlation analysis of bias and variance, i.e., epistemic uncertainty. Therefore, analyzing the epistemic uncertainty, using the established relationship between uncertainty and error items (as shown in Fig. 1), can help to select an appropriate uncertainty quantification method and improve the model performance.

Existing survey papers: There exist several overviews of uncertainty learning techniques in machine learning from different perspectives and emphases (see Table 1). In 2016, Wang and He [21] discussed the challenging issues in analyzing big data and emphasized the importance of modeling uncertainty in improving the performance of the learning models. Subsequently, Hariri et al. [33] surveyed uncertainty learning techniques in big data. They briefly introduced the classical uncertainty measuring techniques in machine learning and categorized them into probability theory, Shannon's entropy, fuzzy set theory, and rough set theory. Recently, Hullermeier and Waegeman [20] emphasized the significance of identifying aleatoric and epistemic uncertainty separately in machine learning. With the popularity of deep learning techniques, most of the recent review papers focused on techniques that are effective for neural network frameworks. For example, Kabir et al. [32] provided a review of uncertainty quantification techniques in neural networks from the concept of prediction intervals. Wang et al. [34] described Bayesian deep learning as a uniform framework that combines deep learning techniques with a paradigm of excellent uncertainty handling capabilities, i.e., probabilistic graphical methods. Jospin et al. [35] organized a handbook from basic statistic concepts to the principle, the learning strategy, and specific algorithms for researchers interested in Bayesian neural networks. They categorized Bayesian methods into Variational inference (VI), Markov Chain

Table 1: Summary of related uncertainty quantification surveys.

Study	Venue	Content
Wang and He (2016) [21]	IEEE Systems, Man, & Cybernetics Magazine	Summarizing challenges of big data and uncertainty-based learning methods.
Kabir et al. (2018) [32]	IEEE access	Discussing from the concept of Prediction Intervals.
Hariri et al. (2019) [33]	Journal of Big Data	Categorizing techniques that handle uncertainty in big data according to various data characteristics.
Wang and Yeung (2020) [34]	ACM Computing Surveys	Outlining Bayesian-based quantification methods from the perspective of PGM.
Jospin et al. (2020) [35]	ACM Computing Surveys	A tutorial specific on Bayesian deep learning.
Hullermeier and Waegeman (2021) [20]	Machine Learning	Emphasizing the important to distinguish different uncertainty.
Abdar et al. (2021) [36]	Information Fusion	Reviewing uncertainty quantification techniques along with their applications.
Gawlikowski et al. (2021) [37]	arXiv	Introducing sources of uncertainty, categorizing uncertainty techniques and reviewing re-calibration techniques .

Monte Carlo (MCMC), and Bayes by backprop. They also discussed approximation techniques in terms of stochastic gradient descent (SGD) dynamics and Monte Carlo dropout (MCD).

Recently, Abdar et al. [36] gave an extensive review of uncertainty quantification methods in deep learning along with their applications. Specifically, they categorized the uncertainty quantification techniques into Bayesian approximation and ensemble learning and discussed the representative models of each category. In addition, they provided open challenges and future research directions associated with uncertainty quantification. While Gawlikowski et al. [37] first identified five sources of uncertainty

in deep learning models. These sources are variability in practical scenarios, error, and noise in measurement tools, errors caused by unknown data samples, errors in model structure and training procedure. Then, they categorized uncertainty learning techniques into single deterministic methods, Bayesian methods, ensemble Methods, test-time augmentation methods. Besides, they reviewed re-calibration techniques in DL. Each of these surveys has provided a comprehensive review of uncertainty learning from different points of view. But none of them include a detailed review of epistemic uncertainty techniques in terms of bias-variance decomposition.

Contributions of the paper: Different from existing surveys, we emphasize the importance of uncertainty analysis in supervised learning models from the perspective of generalization error decomposition. Specifically, the focus is on tracing the epistemic uncertainty according to the decomposed items, i.e., bias and variance. In this paper, we outline the two most representative techniques in supervised learning, e.g., Bayesian and ensemble methods, over the last five years and discuss the properties of each category according to the bias-variance decomposition. In this context, we have collected a total of 138 publications, in which:

- Approximately 70% of the selected articles are published in the last five years, i.e., after 2016; while most of the remaining 30% are high-cited classic articles.
- More than 80% of the selected articles are indexed in Q1 journals, top conferences[1], and high-cited books or thesis

Standing upon these high-quality articles, this work aims to guide researchers interested in tracing the limitation of models from the perspective of uncertainty decomposition, quantification, analysis, and applications. To sum up, the main contributions of this survey include:

- review of the epistemic (model) uncertainty learning techniques in supervised models over the last five years;
- discussion on epistemic uncertainty learning from the perspective of generalization error, i.e., bias and variance decomposition;
- hierarchical categorization of the epistemic uncertainty learning methods along with their representative models and real-world applications;
- elucidation on the main research gaps and suggesting future research directions.

Organization of the paper: As shown in Fig. 2, this review consists of four sections. Section 2, first, provides a review of epistemic uncertainty learning methods.

[1]Top conferences refer to the well-accepted high level conferences, such as NIPS, ICML, AAAI, IJCAI, ICCV, ECCV, CVPR, etc.

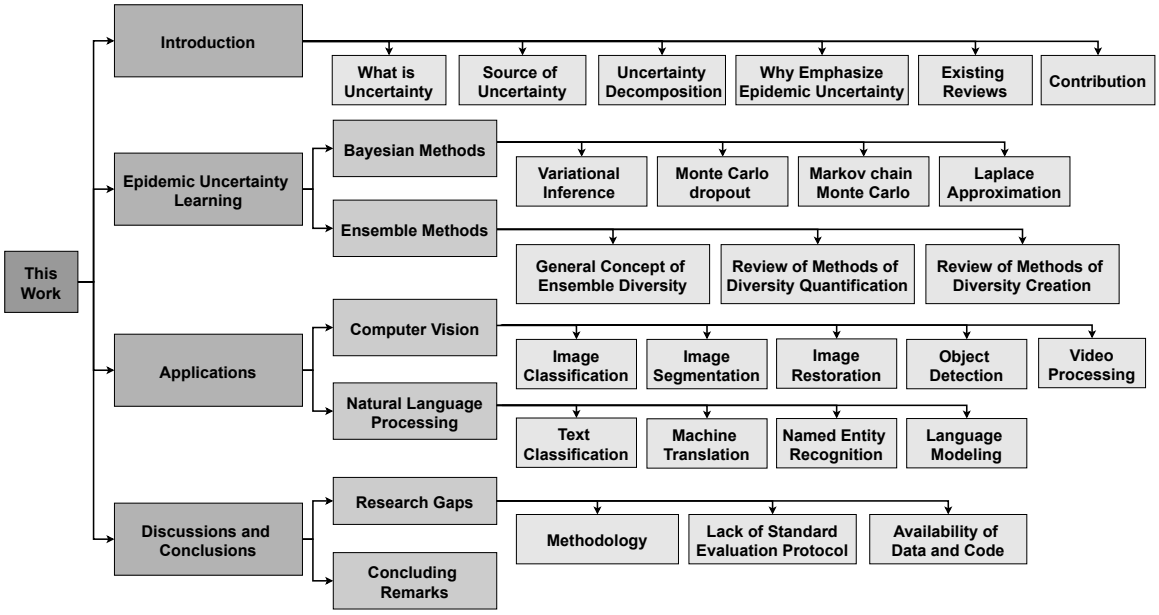

Fig. 2: Organization of this survey.

Specifically, these methods are categorized into Bayesian approximation and ensemble learning. Then, several widely used Bayesian approximation techniques, such as variational inference (VI), Monte Carlo dropout (MCD), Markov Chain Monte Carlo (MCMC), and Laplace approximation (LA), are discussed in detail. Finally, ensemble learning is introduced in terms of its concept and relationship to epistemic uncertainty as well as those related ensemble methods. Section 3 discusses the importance of quantifying uncertainty in supervised learning approaches for several real-world applications such as computer vision and natural language processing. Section 4 presents the research gaps and trends for future research as well as concluding remarks.

2. Review on Epistemic Uncertainty Learning

This section gives a review of the epistemic uncertainty learning techniques in supervised learning. Specifically, we focus on the epistemic uncertainty quantification methods in terms of decomposed items of the generalization error, i.e., bias and variance. According to Fig. 3, the epistemic uncertainty learning methods are grouped into the Bayesian and ensemble methods, as follows:

- **Bayesian methods** formulate epistemic uncertainty as a probability distribution over the model parameters. These methods mainly quantify the variance (as shown in Fig. 1) in order to reduce the generalization error of the learning model. In this context, most studies explore the neural network-based model and Bayesian methods to estimate the epistemic uncertainty caused by the

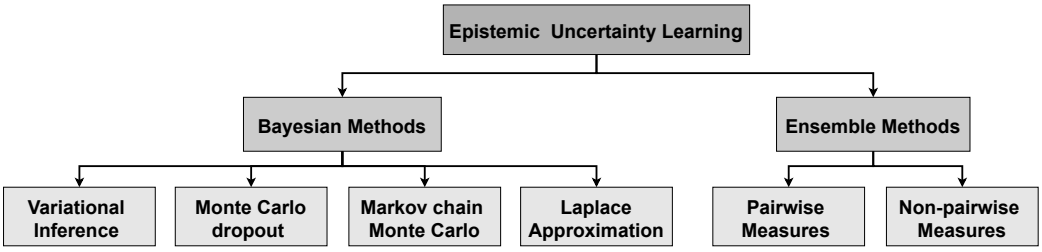

Fig. 3: The taxonomy of epistemic (model) uncertainty learning.

model parameters. Section 2.1 provides a comprehensive review of these techniques.

- **Ensemble methods** train multiple models to produce multiple predictions and then combine their predictions to reach the final output. These methods mainly quantify the variance of the outputs of base models as the epistemic uncertainty to control the complementarity among these base models for improving the ensemble performance. In this context, the reduction of the generalization error can be achieved by reducing the bias or the variance of the ensemble output, depending on the essence of specific ensemble methods. These methods are reviewed in Section 2.2.

In the following subsections, we provide a detailed review and discuss the main properties of each category.

2.1. Bayesian methods

Assume a training data set $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, where $\mathbf{x}_i \in \Re^d$ and $y_i \in \{1, ..., C\}$ indicate the i-th input and its corresponding class, respectively, and C denotes the number of classes. The aim is to learn a function $y = f^{\theta}(\mathbf{x})$ with parameters θ to obtain a desired output. Bayesian modeling aims to capture the epistemic uncertainty by putting distributions over the network weights instead of deterministic network weights, which is known as *marginalisation*. For a given test sample $\mathbf{x}^*$, the distribution over a prediction y^* can be written as [38]:

$$p(y^*|\mathbf{x}^*, D) = \int p(y^*|\mathbf{x}^*, \theta)p(\theta|D)d\theta, \tag{1}$$

where $p(\theta|D)$, which is known as posterior distribution on the model parameters, represents the uncertainty on the model parameters given a training data set D.

Assume $\widehat{\theta}_t$ indicates the parameters of t-th sample from distribution $p(\theta|D)$, the epistemic uncertainty of the model can be quantified via variance term (as shown in Fig. 1):

$$Var(y*) \approx \frac{1}{T}\sum_{t=1}^{T} f^{\widehat{\theta}_t}(\mathbf{x}^*)^{\mathsf{T}} f^{\widehat{\theta}_t}(\mathbf{x}^*) - \mathbb{E}(y*)^{\mathsf{T}}\mathbb{E}(y*), \tag{2}$$

where T is the total number of sampling, which will be explained in the following subsections.

Bayesian methods are easy to implement but difficult to perform inference, because they require to estimate the posterior distribution, i.e., $p(\theta|D)$. Therefore, the marginal probability cannot be computed analytically [26]. In order to obtain the posterior distribution, the Bayes theorem [39] is applied to a given data set D over θ, as follows:

$$p(\theta|D) = \frac{p(D,\theta)p(\theta)}{p(D)}, \tag{3}$$

where $p(D,\theta)$ represents the likelihood that the data samples in D are realization of the distribution predicted by a model with parameter θ, and $p(D)$ is the prior distribution on the model parameters. Scholars have proposed many approximation techniques to estimate posterior distribution. In this survey, we discuss several approximation techniques including variational inference (VI), Monte Carlo dropout (MCD), Markov chain Monte Carlo (MCMC) and Laplace approximation. A detailed review of each category is provided in the following subsections.

2.1.1. Variational inference (VI)

VI [40] has been successfully applied as an approximation technique to many applications of neural networks. It uses a pre-specified distribution $q(\theta)$ to infer the posterior distribution $p(\theta|x,y)$. In other words, VI aims to make $q(\theta)$ to be close to the posterior obtained from the original model through optimizing a set of parameters. To achieve this, the Kullback-Leibler (KL) divergence [41] can be defined as:

$$KL(q||p) = \mathbb{E}_q\left[\log \frac{q(\theta)}{p(\theta|x,y)}\right]. \tag{4}$$

But the KL divergence cannot be directly minimized because of the posterior $p(\theta|x,y)$. Instead, the evidence lower bound (ELBO) can be optimized. As such, the ELBO for a given prior distribution over the model parameters can be written as:

$$L = \mathbb{E}_q\left[\log \frac{p(y|x,\theta)}{q(\theta))}\right]. \tag{5}$$

and for the KL divergence:

$$KL(q||p) = -L + \log\ p(y|x), \tag{6}$$

holds.

Graves et al. [42] introduced a stochastic variational method to reduce the difficulty in inferring analytical solutions of the original VI. They used numerical integration to approximate the expected values. Bayes By Backprop (BBB) [43] is an

extended version of the stochastic variational method [42] to non-Gaussian priors. Specifically, BBB uses an unbiased estimate of gradients to learn a distribution over the network's weights. Kingma et al. [44, 45] reduced the variance of the stochastic gradients by introducing a reparameterization strategy. This strategy can approximate posterior inference in models with continuous latent variables. Rezende and Mohamed [46] used normalizing flow to construct distributions for approximation. This technique applies a sequence of invertible transformations to transfer a simple density to a more complex one. Zeng et al. [47] estimated the epistemic uncertainty in an active learning framework using Bayesian convolutional neural networks (CNNs) with Gaussian approximate VI. They showed that using a few Bayesian layers close to the output layer of CNN can estimate a similar level of uncertainty as with that of the original Bayesian CNNs.

Zhang et al. [48] showed that natural gradient descent [49] with adaptive weight noise can be fitted as a variational posterior to maximize the ELBO. Later, Osawa et al. [50] trained deep networks using a natural gradient VI, namely variational online Gauss-Newton (VOGN) [51], and obtained similar results to that of Adam optimizer by using strategies such as batch normalization and data augmentation. The *stochastic low-rank approximate natural-gradient* (SLANG) [52] is a variant of VI methods that use a structure based on diagonal plus low-rank to compute the Gaussian approximations. In addition, SLANG uses the back-propagated gradients of the network log-likelihood to build a covariance, which enables the model a faster estimation than mean-field methods. Heo et al. [53] proposed *uncertainty-aware attention*, which uses VI with dropout sampling, to compute the epistemic uncertainty together with the heteroscedastic uncertainty in predicting time-series data.

2.1.2. Monte Carlo dropout (MCD)

Monte Carlo (MC) [54] is another approximation technique that has been widely used for estimating the posterior distribution, but it is computationally expensive and slow. Gal devised MC dropout (MCD) [55, 38] to alleviate these issues. MCD integrates dropout [56], which is an effective technique for tackling overfitting problems in deep models, as a regularization term to estimate the prediction uncertainty. During learning, dropout randomly (with a certain probability p) drops some model units to avoid excessive co-tuning. MCD uses the mean of N models, $f_{\theta_1}, ..., f_{\theta_N}$, parametrized by $\theta_1, ..., \theta_N$ to approximate outcome based on the posterior estimation of the weights as follows [55]:

$$y^* \approx \frac{1}{N} \sum_{i=1}^{N} y_i^* = \frac{1}{N} \sum_{i=1}^{N} f_{\theta_i}(x^*). \tag{7}$$

Later, Gal et al. [57] introduced a new variant of dropout, called concrete dropout, which uses gradient methods instead of grid search to tune the dropout probability. This leads to a calibrated uncertainty estimate in large models. Mokhoti and Gal [58]

integrated MCD and concrete dropout as inference techniques into the *DeepLab-v3+* [59] structure sense segmentation. In addition, they introduced a new metric, namely *mutual information*, to estimate epistemic uncertainty by computing the mutual information between a predictive distribution and posterior over the model weights. The single-shot MCD [60] analytically approximates the expected value and variance of the MCD for each layer of the fully connected network. This model requires less computational time as compared with that of the MCD. Study [61] conducted an empirical study to show how epistemic uncertainty is affected when the observing condition is changed using MCD.

Since the proposal of MCD, many scholars have applied it to estimate epistemic uncertainty. For example, Abdar et al. [62, 63] applied MCD to tackle uncertainty during skin cancer image classification. Studies [64] and [6] integrated MCD into CNN to estimate the epistemic uncertainty for segmentation and lesion detection in medical images. Loquercio et al. [65] computed the epistemic uncertainty in robotics by combining Bayesian belief networks with MCD. Bertoni et al. [66] estimated the epistemic uncertainty for Monocular 3D Pedestrian Localization using MCD. In [67], MCD is used to estimate the epistemic uncertainty in an encoder-decoder framework with long-short-term-memory (LSTM) for time series forecasting and anomaly detection using Uber data. Xiao and Wang [11] utilized MCD to estimate the epistemic uncertainty for natural language processing tasks.

2.1.3. Markov chain Monte Carlo (MCMC)

MCMC is another popular technique to approximate inference and represents epistemic uncertainty. It first samples from arbitrary distributions and then performs a stochastic transition governed by the current state and the desired distribution, e.g., true posterior. In other words, MCMC starts with generating samples in an iterative and Markov chain fashion. Markov chain is a distribution over random variables that undergoes a transition from one state to another one in the space state. At each iteration, the model selects samples based on pre-specified rules. This process is iterated T times. Finally, the desired distribution is approximated using the generated samples. It aims to sample for a set of independent observations $\mathbf{x} \in D$ from the posterior distribution θ [68]:

$$p(\theta|D) \propto \exp(-U(\theta)), \tag{8}$$

where U is the potential energy function defined by:

$$U = -\sum_{x \in D} log\ p(x|\theta) - log\ p(\theta). \tag{9}$$

Hamiltonian (hybrid) MC (HMC) [54, 69] is the first one that involves using the MCMC sampling technique for Bayesian neural networks. It explores the state space based on the Metropolis-Hastings framework instead of a random-walk strategy to sample from θ. As such, it introduces a set of auxiliary momentum variables, denoted

by r, from a Hamiltonian system. In order to sample from $p(\theta|D)$, HMC generates samples from a joint distribution of (θ, r) as follows:

$$\pi(\theta, r) \propto \exp\left(- U(\theta) - \frac{1}{2} r^T \ M^{-1} r \right), \tag{10}$$

where M, which is a mass matrix and often set to the identity matrix, together with r indicates a *kinetic energy* term. The Hamiltonian function in given by:

$$H(\theta, r) = U(\theta) + \frac{1}{2} r^T \ M^{-1} r. \tag{11}$$

The Hamiltonian dynamics is simulated by HMC to generate samples, as follows:

$$\begin{cases} d\theta = M^{-1} r \ dt \\ dr = - \bigtriangledown U(\theta) \ dt \end{cases}. \tag{12}$$

Despite the success of HMC, it requires processing all data samples at each iteration, which is computationally expensive specifically for large data sets. To alleviate this issue, many algorithms have attempted to use a mini-batch strategy. In this regard, Welling and Teh [70] proposed stochastic gradient descent (SGD) HMC method that combines SGD with first-order Langevin dynamics. Later, Chen et al. [68] proved that using second-order Langevin dynamics can explore the space of solutions and provide good generalization. In addition, they added friction into the SGD-HMC to update momentum and evaluated the impact of the noisy gradient. Teye et al. [71] showed that training deep models with batch normalization is equal to that of estimating the inference in Bayesian networks. Chandra et al. [72] proposed Bayesian graph deep learning techniques that use MCMC samples with Langevin-gradient. Mandt et al. [73] used SGD with a constant learning rate (constant SGD) to simulate the Markov chain with a stationary distribution and showed that constant SGD can approximate the posterior inference. Cyclical stochastic gradient MCMC (SG-MCMC) [74] used a cyclical stepsize schedule to better approximate posterior distributions. However, using a mini-batch strategy, that employs a small set of samples at each iteration, adds noise to the network and increases its uncertainty. To alleviate this, Luo et al. [75] used Nosé-Hoover thermostats [76] to deal with the generated noise. The resulting method is called *thermostat-assisted continuously tempered HMC*.

Maddox et al. [77] introduced stochastic weight averaging Gaussian (SWA-Gaussian) to represent uncertainty and calibrate deep models. Specifically, SWA-Gaussian uses SWA [78] to compute the mean of SGD iterates with a high constant learning rate in order to improve the generalization in deep models. In addition, the Gaussian posterior approximation over the model weights is approximated by using mean SWA and computing a low-rank plus diagonal approximation to the covariance of the iterates. Garg and Awate [5] proposed a perfect/exact MCMC for generic Markov random

fields to compute the uncertainty in multi-label segmentation. Specifically, they combined two schemes, namely coupling from the past [79] and bounding-chain [80], to propose perfect-sample label images. Hernández et al. [81] combined dropout and HMC to improve the predictive uncertainty in classification problems. Akkoyun et al. [82] applied MCMC into a Bayesian framework to predict maximum aneurysm diameter. In addition, Cai et al. [83] developed proximal MCMC techniques to estimate uncertainty in radio interferometric imaging.

2.1.4. Laplace approximation

As another powerful approximating method, Laplace approximation tackles the problem of representing a complex posterior over the parameters of neural networks by assuming it as a Gaussian distribution [84]. Different from variational approximation methods, Laplace approximation is a local approximation technique that pays more attention to the trend around the mode of the posterior distribution. As described in [85], the expectation μ of Gaussian distribution $q(\theta)$ is the extreme point θ^* of posterior distribution $p(\theta|D)$. Thus, μ is determined by the first derivative of $p(D,\theta)$ which meet the condition $p(\theta|D) \propto p(D,\theta)$, while the covariance matrix $\mathbf{\Sigma}$ is obtained by the second-order Taylor expansion of $\ln p(D,\theta)$ centering on θ^*:

$$\ln p(D,\theta) \approx \ln p(D,\theta^*) - \frac{1}{2}(\theta-\theta^*)^{\mathsf{T}}\mathbf{H}(\theta-\theta^*), \tag{13}$$

where $\mathbf{H}$ is Hessian matrix which defined as:

$$\mathbf{H} = -\frac{\partial^2 \ln p(D,\theta)}{\partial\theta^2}\bigg|_{\theta=\theta^*}. \tag{14}$$

Then, the posterior $p(\theta|D)$ is approximated as Gaussian $q(\theta)$ with covariance matrix $\mathbf{\Sigma} = \mathbf{H}^{-1}$:

$$p(\theta|D) \approx q(\theta) \sim N(\theta|\theta^*, \mathbf{H}^{-1}). \tag{15}$$

Unfortunately, it is infeasible to compute the Hessian matrix for deep neural networks with a significant number of parameters. Relatively, constructing a diagonal matrix in curvature approximating for a neural network is more calculable and efficient. Kirkpatrick et al. [86] used diagonal Laplace approximations to enhance the capability of deep neural networks for sequentially learning tasks by preserving the weights important for previous tasks. Subsequently, Ritter et al. [87] first pointed out the limitation of diagonal approximation when some weights exhibit high covariance, then suggested the effectiveness of Kronecker Factorization for acquiring covariance in Laplace approximation and successfully applied in learning online scenarios [88].

More recently, Lee et al. [89] developed a sparsification technique using a low-rank approximation to demonstrate the effectiveness of scaling Laplace approximation to large-sized data sets (e.g., ImageNet) and architectures. Schillings et al. [90] discussed

the convergence of Laplace approximation in Hellinger distance. Margossian et al. [91] derived an adjoint method to promote the computation of Monte Carlo with an embedded Laplace approximation in order to marginalize out weights. Daxberger et al. [92] obtained posteriors by performing inference over a small subset of model weights and outlined the procedure for scaling the linearized Laplace approximation to large neural network models within the framework of subnetwork inference. A new idea, L2M [93], estimated uncertainty by expanding Laplace approximation with gradient raw second moment estimation in optimizers.

2.2. Ensemble methods

Ensemble learning is generally aimed at training multiple models that are combined to make a final prediction. In a regression context, simple averaging is a commonly used way of combining multiple models $f_i : X \to Y$ for $i = 1, \ldots, M$ in an ensemble $f_{ens} : X \to Y$, as illustrated below:

$$f_{ens}(x) = \frac{1}{M} \sum_{i=1}^{M} f_i(x). \tag{16}$$

In a classification context, majority voting is a commonly used rule of fusing multiple classifiers to finally output a class $c = 1, 2, \ldots, k$, as illustrated below:

$$f_{ens}(x) = \arg\max_{c} \sum_{i=1}^{M} I(f_i(x) = c). \tag{17}$$

In general, it often appears that the outputs of multiple models in an ensemble are different, where the variance of the outputs is considered as an indicator of epistemic uncertainty in a prediction [20, 94, 95]. On the other hand, the epistemic uncertainty is also referred to as ensemble ambiguity (diversity), which is viewed as a key factor of successful ensemble learning [96, 97]. In this section, we will introduce the general concept of ensemble diversity and analyze the importance of the diversity in terms of improving the ensemble performance. Moreover, we will provide a review of those existing methods of diversity quantification and creation.

2.2.1. General Concept of Ensemble Diversity

Ensemble diversity is generally related to the generalization error of an ensemble. In a regression context, the generalization error can be decomposed through two well-known schemes, namely, ambiguity decomposition [98] and bias-variance decomposition [99].

In terms of ambiguity decomposition, given a weighted averaging ensemble (illustrated in Eq. (18)), the ensemble error $(f_{ens} - y)^2$ can be decomposed into two terms, i.e., the average error of the based models $\frac{1}{M} \sum_{i=1}^{M} w_i (f_i - y)^2$ and the ensemble ambiguity (diversity) $\frac{1}{M} \sum_{i=1}^{M} w_i (f_i - f_{ens})^2$, as illustrated in Eq. (19).

$$f_{ens}(x) = \sum_{i=1}^{M} w_i f_i(x), \tag{18}$$

where w_i is the weight of each base model f_i with the constraints $0 \leq w_i \leq 1$ and $\sum_{i=1}^{M} w_i = 1$, i.e., f_{ens} is essentially a convex combination of the M base models.

$$(f_{ens} - y)^2 = \frac{1}{M}\sum_{i=1}^{M} w_i (f_i - y)^2 - \frac{1}{M}\sum_{i=1}^{M} w_i (f_i - f_{ens})^2. \tag{19}$$

According to Eq. (19), it is straightforward to derive that the ensemble error is guaranteed to be less than or equal to the average error of the base models, i.e., the higher the diversity among the base models is created, the larger the error reduction would be achieved. However, the increase of the diversity may also cause the increase of the average error of the base models [100], so it is necessary to get the reasonable trade-off between the diversity and the average error.

As discussed in [100], the ambiguity decomposition does not take into account the possible changes of the training data distribution or the initialized weights distribution. However, it is essential to measure effectively the expected error on unseen data given a specific distribution of training data or initialized weights. From this point of view, the bias-variance decomposition scheme is considered as a useful tool for analyzing the generalization error of an ensemble, given that this scheme exactly takes into account the above mentioned changes of distributions.

The general formulation of the bias-variance decomposition is shown in Fig. 1 in Section 1. Based on this formulation, three concepts have been defined in [100], namely, the averaged bias, the averaged variance and the averaged co-variance of the M base models, as illustrated below:

$$\overline{bias} = \frac{1}{M}\sum_{i=1}^{M} (E\{f_i\} - y), \tag{20}$$

$$\overline{var} = \frac{1}{M}\sum_{i=1}^{M} E\{(f_i - E\{f_i\})^2\}, \tag{21}$$

$$\overline{covar} = \frac{1}{M(M-1)}\sum_{i=1}^{M}\sum_{j\neq i} E\{(f_i - E\{f_i\})(f_j - E\{f_j\})\}. \tag{22}$$

According to Eq. (20)-(22), we can obtain the bias-variance-co-variance decomposition of the mean square error of an ensemble f_{ens}, as shown below:

$$E\{(f_{ens} - y)^2\} = \overline{bias}^2 + \frac{1}{M}\overline{var} + (1 - \frac{1}{M})\overline{covar}. \tag{23}$$

Eq. (23) indicates that the mean square error of an ensemble f_{ens} generally depends on the correlation between those base models, where the correlation is quantified through the third term (the averaged co-variance of the base models). Therefore, it is expected to decrease the co-variance, without affecting the bias and variance [100].

Moreover, the connection between the ambiguity decomposition and the bias-variance-co-variance decomposition was disclosed in [100]. In particular, according to Eq. (19), the ensemble error $(f_{ens} - y)^2$ can be decomposed into the average error of M base models $\frac{1}{M}\sum_{i=1}^{M} w_i(f_i - y)^2$ and the ensemble ambiguity $\frac{1}{M}\sum_{i=1}^{M} w_i(f_i - f_{ens})^2$. While assuming that the base models are equally weighted for simplicity, the right hand side of Eq. (19) can be substituted into the left hand side of Eq. (23) to obtain a new formulation as below:

$$E\{\frac{1}{M}\sum_{i=1}^{M}(f_i - y)^2 - \frac{1}{M}\sum_{i=1}^{M}(f_i - f_{ens})^2\} = \overline{bias}^2 + \frac{1}{M}\overline{var} + (1 - \frac{1}{M})\overline{covar}. \quad (24)$$

Based on Eq. (24), the following formulations can be obtained after some derivations [100]:

$$\begin{aligned} E\{\frac{1}{M}\sum_{i=1}^{M}(f_i - f_{ens})^2\} &= \frac{1}{M}\sum_{i=1}^{M} E\{(f_i - E\{f_i\})^2\} - E\{(f_{ens} - E\{f_{ens}\})^2\} \\ &= \overline{var} - var(f_{ens}) = \overline{var} - \frac{1}{M}\overline{var} - (1 - \frac{1}{M})\overline{covar} \end{aligned} \quad (25)$$

$$E\{\frac{1}{M}\sum_{i=1}^{M}(f_i - y)^2\} = \frac{1}{M}\sum_{i=1}^{M}(E\{f_i\} - y)^2 + \frac{1}{M}\sum_{i=1}^{M} E\{(f_i - E\{f_i\})^2\} = \overline{bias}^2 + \overline{var} \quad (26)$$

It can be seen from Eq. (25)-(26) that the variance term $\overline{var}$ relates to both the average error of base models and the ensemble ambiguity, so the subtraction of Eq. (25) from Eq. (26) gets Eq. (23) back and cancels out the variance term $\overline{var}$ (not $\frac{1}{M}\overline{var}$). Moreover, the fact that the variance term $\overline{var}$ appears in both Eq. (25) and Eq. (26) indicates that it is generally difficult to simply maximize the ensemble ambiguity without affecting the bias term $\overline{bias}$ [100, 101].

The above two error decomposition schemes were generally designed for regression problems and can not be directly applied to classification tasks. In terms of the ambiguity decomposition, Eq. (19) derived in a regression setting can be transformed into Eq. (27) [102] to suit a classification task, while assuming that the M base classifiers are fused by combining the class probability values estimated by these classifiers through using the product rule [101]. In particular, the KL divergence

$D_{KL}(y||f_{ens})$ of the ensemble f_{ens} from the target distribution y of class probability is defined as the ensemble error, which can be decomposed into two terms, namely, the average KL divergence $\frac{1}{M}\sum_{i=1}^{M} D_{KL}(y||f_i)$ of the class probability estimates of base classifiers from the target distribution y and the ambiguity $\frac{1}{M}\sum_{i=1}^{M} D_{KL}(f_{ens}||f_i)$ of the ensemble f_{ens}.

$$D_{KL}(y||f_{ens}) = \frac{1}{M}\sum_{i=1}^{M} D_{KL}(y||f_i) - \frac{1}{M}\sum_{i=1}^{M} D_{KL}(f_{ens}||f_i). \tag{27}$$

According to Eq. (27), the KL divergence $D_{KL}(y||f_{ens})$ of the ensemble f_{ens} from the target distribution y is guaranteed to be less than or equal to the average KL divergence $\frac{1}{M}\sum_{i=1}^{M} D_{KL}(y||f_i)$ of the class probability estimates of the M base classifiers, i.e., the higher the ambiguity $\frac{1}{M}\sum_{i=1}^{T} D_{KL}(f_{ens}||f_i)$ of ensemble f_{ens} is created, the higher the performance improvement would be achieved.

However, the combination of multiple classifiers can be achieved in various ways, e.g., majority vote and average of class probability distributions, which indicates that Eq. (27) does not provide a general formulation of ensemble diversity for classification tasks.

In terms of the bias-variance decomposition, as discussed in [101], Eq. (23) was derived only for regression problems and similar results cannot be obtained for classification tasks. Therefore, Eq. (23) can not be used as a general formulation of ensemble diversity either. Those methods of diversity quantification in a classification context will be introduced in Section 2.2.2. Moreover, there is not yet a formally accepted definition of the diversity term [101, 103, 96, 97], so existing methods of diversity creation were designed heuristically using different definitions and the methods will be reviewed in Section 2.2.3.

2.2.2. Review of Methods of Diversity Quantification

In a classification context, if the ensemble prediction is made by averaging the class probabilities estimated by M classifiers, then the ensemble diversity can be measured based on Tumer and Ghosh's framework [104, 105]. In particular, suppose that each class c has a true posterior probability $P_d(c|x)$ and another posterior probability $P_{e_i}(c|x)$ estimated by a classifier f_i, given a one-dimensional feature vector x. In this context, the classification error can be decomposed into the Bayes error and the added error, where the Bayes error is irreducible and the added error $\eta_{e_i}(c|x)$ results from the incorrect estimation of the class posterior probability as illustrated below:

$$P_{e_i}(c|x) = P_d(c|x) + \eta_{e_i}(c|x). \tag{28}$$

If it is assumed that the errors of posterior probability estimation on two classes a and b are independent and identically distributed random variables [105] with zero mean and variance $\sigma^2_{\eta_i}$, then the expected added error of classifier f_i in distinguishing the two classes can be defined as shown below:

$$E_{add,i} = \frac{\sigma^2_{\eta_i}}{P'_d(a|x) - P'_d(b|x)}, \tag{29}$$

where $P'_d(a|x)$ and $P'_d(b|x)$ are the derivatives of the true posterior probabilities of classes a and b. In the case of combing the posterior probabilities estimated by M classifiers, the expected added error can be measured in the way as illustrated below [105]:

$$E^{ensemble}_{add} = E_{add}(\frac{1 + \delta(M-1)}{M}). \tag{30}$$

In Eq. (30), δ is a correlation coefficient used for measuring the correlation among the estimation errors made by M base classifiers for each class and is thus a way of quantifying the ensemble diversity. If the estimation errors of the M classifiers are independent, i.e., $\delta = 0$, then the expected added error of the ensemble would be $\frac{1}{M}$ as same as the added error of each of the M base classifiers (that are assumed to have the same estimation error). However, if the estimation errors of the M base classifiers are perfectly correlated, i.e., $\delta = 1$, then the expected added error of the ensemble would be the same as the added error of each base classifier. Moreover, if the estimation errors of the M base classifiers are negatively correlated, i.e., $\delta < 0$, then the expected added error of the ensemble can be reduced even more in comparison with the amount of the error reduction in the case of $\delta = 0$ [96].

In addition to the average rule of fusion, i.e. averaging the class probabilities estimated by multiple classifiers, majority voting is also a popular rule of combining classifiers. Since the outputs of the M classifiers in a majority vote ensemble are not numeric, the correlation coefficient δ used in Eq. (30) can not be applied directly. Instead, some researchers have tried to define the classification error diversity qualitatively. For example, a scheme has been suggested in [106] to classify error diversity into four levels as follows:

- Level 1: At most one of the base classifiers in an ensemble makes incorrect classification for each instance.
- Level 2: The majority of the base classifiers in an ensemble make correct classification for each instance.
- Level 3: At least one of the base classifiers in an ensemble make correct classification for each instance.
- Level 4: All of the base classifiers in an ensemble make incorrect classification for each instance.

Furthermore, a decomposition of the majority vote error E_{maj} into the average error E_{avg} of the base classifiers, good and bad diversity was introduced in [102], where good diversity has a positive impact on the error reduction and bad diversity results in a negative impact. In this context, Level 1 and Level 2 diversity would be classified as good ones whereas Level 3 and Level 4 diversity would be classified as bad ones. In the setting of binary classification, i.e., $y \in \{+1, -1\}$, the majority vote error decomposition is shown as below:

$$E_{maj} = \sum_{x} E_{avg}(x) - \sum_{x} y(x)\bar{f}(x)\frac{1}{M}\sum_{i=1}^{M} Dis_i(x), \tag{31}$$

where the disagreement Dis_i between a base classifier f_i and the ensemble $\bar{f}$ is measured using:

$$Dis_i(x) = \frac{1}{2}(1 - f_i(x)\bar{f}(x)). \tag{32}$$

In Eq. (31), the sign of $y(x)\bar{f}(x)$ essentially reflects whether the ensemble classification is correct or not, i.e., $y(x)\bar{f}(x) = +1$ represents correct classification and $y(x)\bar{f}(x) = -1$ indicates incorrect classification. Therefore, Eq. (31) can be rewritten as below:

$$E_{maj} = \sum_{x} E_{avg}(x) - \sum_{x+} \frac{1}{M}\sum_{i=1}^{M} Dis_i(x) + \sum_{x-} \frac{1}{M}\sum_{i=1}^{M} Dis_i(x). \tag{33}$$

where the second term $\sum_{x+} \frac{1}{M}\sum_{i=1}^{M} Dis_i(x)$ denotes good diversity and the third term $\sum_{x-} \frac{1}{M}\sum_{i=1}^{M} Dis_i(x)$ denotes bad diversity.

Table 2: Contingency table for classifiers f_i and f_j

	f_i correct(1)	f_i incorrect(0)
f_j correct(1)	N^{11}	N^{10}
f_j incorrect(0)	N^{01}	N^{00}

Those representative methods of the diversity quantification have been analysed in [96, 101], which are put into two main categories, namely, pairwise measures and non-pairwise measures. In particular, pairwise measures are generally designed by involving calculation based on a contingency table shown in Table 2 for a pair of classifiers f_i and f_j, whereas non-pairwise measures generally involve counting the number $l(x)$ of classifiers that correctly classify sample x and calculating the relevant probability $P(l(x))$.

Table 3: Summary of Diversity Measures

Name	Symbol	↑/↓	P	Equation	Range
Q-statistic	Q	(↓)	Y	$Q_{i,j} = \frac{N^{11}N^{00} - N^{10}N^{01}}{N^{11}N^{00} + N^{10}N^{01}}$	$[-1, 1]$
correlation coefficient	ρ	(↓)	Y	$\rho_{i,j} = \frac{N^{11}N^{00} - N^{10}N^{01}}{\sqrt{(N^{11}+N^{10})(N^{01}+N^{00})(N^{11}+N^{01})(N^{10}+N^{00})}}$	$[-1, 1]$
disagreement measure	dis	(↑)	Y	$dis_{i,j} = \frac{N_{01}+N_{10}}{N^{11}+N^{10}+N^{01}+N^{00}}$	$[0, 1]$
double-fault measure	DF	(↓)	Y	$DF_{i,j} = \frac{N_{00}}{N^{11}+N^{10}+N^{01}+N^{00}}$	$[0, 1]$
entropy	ENT	(↑)	N	$ENT = \frac{1}{\lvert DS\rvert} \sum_{x \in DS} \frac{1}{M - \lfloor M/2 \rfloor} min\{l(x), M - l(x)\}$	$[0, 1]$
Kohavi-Wolpert variance	KW	(↑)	N	$KW = \frac{1}{\lvert DS\rvert M^2} \sum_{x \in DS} l(x)(M - l(x))$	$[0, 1]$
inter-rater agreement	IRA	(↓)	N	$IRA = 1 - \frac{\frac{1}{M} \sum_{x \in DS} l(x)(M-l(x))}{\lvert DS\rvert (M-1)\bar{p}(1-\bar{p})}$	$[0, 1]$
difficulty measure	DM	(↓)	N	$DM = Var(\frac{l(x)}{M})$	$[0, 1]$
generalized diversity	GD	(↑)	N	$GD = 1 - \frac{\sum_{l(x)=1}^{M} \frac{M-l(x)}{M} \frac{(M-l(x)-1)}{(M-1)} p(l(x))}{\sum_{l(x)=1}^{M} \frac{M-l(x)}{M} p(l(x))}$	$[0, 1]$
coincident failure diversity	CFD	(↑)	N	$CFD = \begin{cases} 0, p(0) = 1.0; \\ \frac{1}{1-p(0)} \sum_{l(x)=1}^{M} \frac{l(x)}{M-1} p(l(x)), p(0) < 1 \end{cases}$	$[0, 1]$

Those popularly used pairwise measures include the Q-statistic Q, the correlation coefficient ρ, the disagreement measure dis and the double-fault measure DF. Representative non-pairwise measures of diversity include entropy ENT, Kohavi-Wolpert variance KW, measurement of inter-rater agreement IRA, difficulty measure DM, generalized diversity GD and coincident failure diversity CFD. More details of these diversity measures are summarised in Table 3.

In terms of Q-statistics, the value of $Q_{i,j}$ is ranged in $[-1, 1]$ and is expected to be 0 for two statistically independent classifiers f_i and f_j. While the two classifiers provide the same outputs correctly, the value of $Q_{i,j}$ tends to be positive [96]. In contrast, if the two classifiers make incorrect classifications on different instances, the value of $Q_{i,j}$ would be rendered negative [96]. Therefore, it can be concluded that the lower the value of $Q_{i,j}$ resulting from a pair of classifiers f_i and f_j, the higher the diversity between f_i and f_j is created.

The value of the correlation coefficient $\rho_{i,j}$ is also ranged in $[-1, 1]$. According to formulations of $Q_{i,j}$ and $\rho_{i,j}$, it is straightforward to identify that the values of $Q_{i,j}$ and $\rho_{i,j}$ always obtain the same sign but $|\rho_{i,j}| \geq |Q_{i,j}|$ [96, 101].

According to the formulations of the disagreement measure $dis_{i,j}$ and the double-fault measure $DF_{i,j}$, it is easy to see that both $dis_{i,j}$ and $DF_{i,j}$ are ranged in $[0, 1]$. However, $dis_{i,j}$ and $DF_{i,j}$ aim at measure of diversity in different levels (according to the scheme suggested in [106] for classifying the diversity into four different levels), i.e., $dis_{i,j}$ shows the percentage of samples that are classified incorrectly by only one of the two base classifiers f_i and f_j, whereas $DF_{i,j}$ is the proportion of samples that are mis-classified by both classifiers.

While it is needed to measure the diversity among multiple classifiers, the above-introduced pairwise measures can be used by averaging the values obtained for all

those pairs of classifiers. For example, the Q statistic can be used for diversity measure through averaging the values of Q obtained for all classifier pairs, as shown below:

$$Q_{avg} = \frac{2}{M(M-1)} \sum_{i=1}^{M-1} \sum_{k=i+1}^{M} Q_{i,j}. \tag{34}$$

All the non-pairwise measures are ranged in $[0, 1]$. In the formulations of ENT, KW and IRA, $|DS|$ represents the number of samples in a data set DS and $l(x)$ denotes the number of classifiers that correctly classify sample x. In addition, $\bar{p}$ used in the formulation of IRA denotes the average accuracy of base classifiers.

In terms of the entropy ENT, the minimum value is obtained when all the M classifiers provide the same outputs and the maximum value is obtained when $\lfloor M/2 \rfloor$ classifiers consistently classify sample x as one class and the other $M - \lfloor M/2 \rfloor$ classifiers consistently output another class for x. As emphasized in [96], the Kohavi-Wolpert variance KW is correlated to the average disagreement measure dis_{avg} by a coefficient $\frac{M-1}{2M}$, i.e., $KW = \frac{M-1}{2M} dis_{avg}$. Moreover, the inter-rater agreement IRA is correlated to both KW and dis_{avg} [96], i.e., $IRA = 1 - \frac{M}{|DS|(M-1)\bar{p}(1-\bar{p})} KW = 1 - \frac{1}{2\bar{p}(1-\bar{p})} Dis_{avg}$. In terms of the difficult measure DM, it is generally expected that each instance is difficult for some classifiers but is easy for the other classes to encourage the ensemble diversity [96, 101]. The minimum of DM is obtained while each instance is easy for the majority of the base classifiers, and the maximum of DM is obtained while each instance is either easy or difficult for all the base classifiers. The generalized diversity GD is designed based on the argument that the incorrect output of one classifier is always accompanied by the correct output of another classifier for maximizing the diversity [107]. Furthermore, the coincident failure diversity CFD is defined as a modification of GD [107], which expects that each instance can be classified correctly by some of the base classifiers. While each instance is classified incorrectly by at most one base classifier, the maximum of CFD would be reached, i.e., the highest level of diversity is reached according to the scheme suggested in [106] for identifying the level of diversity.

More recently, Yin et al presented a formulation of diversity learning in [108] as shown below:

$$\min_{w} f_{loss}(w) - \beta f_{diversity}(w) \qquad s.t. \qquad w \geq 0 \tag{35}$$

where the diversity is treated as a regularization term, β is used as a control parameter for the diversity regularization and w represents the model parameter. A formulation of sparsity learning was also presented in [108] for the purpose of ensemble pruning.

Based on the work presented in [108], more studies have been conducted later on by using diversity as a regularization term [97, 109, 110, 111, 112, 113, 114], such that the ensemble accuracy and diversity can be optimized simultaneously. In particular, Cavalcanti et al [97] proposed to combine different pairwise measures of diversity for

ensemble pruning, while the genetic algorithm is used to optimize the combined diversity to obtain several candidate ensembles that are evaluated using the validation data for selecting the final ensemble. Ahmed et al [109] made an empirical investigation on whether combining the ensemble accuracy with several popular diversity measures is a better evaluation function than using only the accuracy in the setting of ensemble pruning. Dai et al pointed out in [110] that accuracy and diversity are closely related to each other and should be considered simultaneously for ensemble pruning. Accordingly, they proposed three new measures for ensemble pruning, namely, Simultaneous Diversity & Accuracy, Diversity-Focused-Two and Accuracy-Reinforcement. Dvornik et al [111] proposed to encourage the ensemble diversity by enabling the classifiers to output consistently the highest probability for the ground truth class label and to rank inconsistently the other classes by making different classes obtain the second-highest probability (or other lower-ranked probability). Zhang et al [112] proposed to construct a diversified ensemble layer for combining multiple neural networks as individual modules, while the cross entropy loss of each individual module and the diversity among different modules are optimized simultaneously. Bian et al [113] formulated the relationship between the diversity and the ensemble performance in the context of the theorem of margin and generalization and proposed two diversity-driven pruning methods to utilize the formulated relationship, leading to the enhancement of diversity and the reduction of the ensemble size without much loss of performance. Wu et al [114] revised those representative diversity measures and introduced focal model based measures of diversity for improving further the correlation between the diversity and the prediction accuracy. Overall, all of the above-reviewed works indicate that effective selection and combination of diversity measures would be essential, such that simultaneous optimization of the ensemble accuracy and diversity can be achieved effectively.

2.2.3. Review of Methods of Diversity Creation

In general, ensemble diversity can be created using various types of methods. In this section, we provide a detailed review of diversity creation methods that fall in the category of data input manipulation. We also briefly introduce other methods in the following categories: data output manipulation, manipulation of model architectures, differentiation of starting points in hypothesis space and diversification of learning strategies.

In the setting of data input manipulation, some popularly used methods include Bagging [115], Random Subspace [116] and Boosting [117]. Bagging involves training M independent classifiers on M different sample sets drawn by random sampling from the original training set with replacement over M iterations. In this setting, the diversity is created heuristically through diversifying the training samples, which contributes to the variance reduction in the context of bias-variance trade-off [118]. Moreover, two variants of Bagging, namely, Dagging [119, 120] and Wagging [118], were developed by setting different ways of diversifying the training samples. In

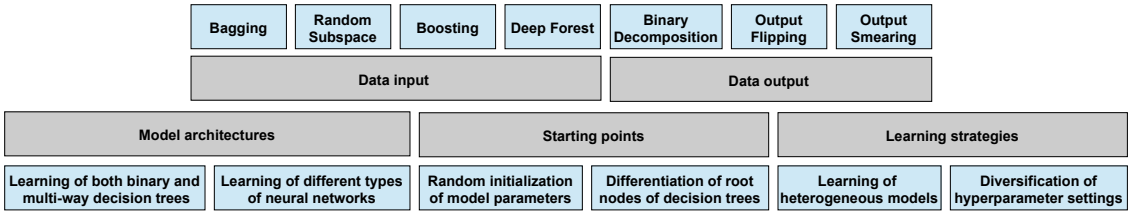

Fig. 4: Methods of diversity creation in Ensemble Learning.

particular, Dagging involves drawing M equal-sized sample sets by partitioning the original training set into M disjoint training subsets, whereas Wagging involves learning each of the M base classifiers from the entire training set but each training sample is assigned a random weight.

Random Subspace can be viewed as a way of diversity creation through feature sampling, which involves training M independent classifiers on M different feature subsets produced by random sampling of features from the full feature set without replacement. Therefore, the Random Subspace method aims at creating diversity heuristically through diversifying the features. A variant of Random Subspace, which is referred to as 'Attribute Bagging' [121], was designed to require a suitable subspace size to be set as a hyper-parameter for drawing M feature subsets. However, in the setting of Random Subspace, all the base classifiers are learned from the entire training set without diversity creation through manipulation of training samples. In order to better enhance the diversity among base classifiers through the combination of Bagging and Random Subspace [122], the Random Forest method [123] has been developed and used as a powerful decision tree ensemble approach [124].

In contrast to Bagging and Random Subspace that create diversity heuristically leading to independent classifiers, the Boosting approach aims at creating diversity explicitly by training one classifier that aims at correcting the errors resulting from the classifier trained at the previous iteration, i.e., training negatively correlated base classifiers. In particular, two popular methods of Boosting are referred to as 'Adaptive Boosting' (AdaBoost) [125] and 'Gradient Boosting' [126]. The former method involves diversity creation through re-weighting samples at each iteration i of learning a base classifier f_i. In other words, at the end of each iteration i, the weight of each correctly classified sample is decreased and the weight of each misclassified sample is increased, so the learning of classifier f_{i+1} will focus more on those misclassified samples. The re-weighting of each sample e is operated in the way shown below:

$$\omega_e^{i+1} = \frac{\omega_e^i exp(-\alpha_i y_e f_i(x_e))}{Z_i} = \left\{ \begin{array}{ll} \omega_e^i exp(-\alpha_i), & \text{if } y_e = f_i(x_e) \\ \omega_e^i exp(\alpha_i), & \text{if } y_e \neq f_i(x_e) \end{array} \right\}. \tag{36}$$

In Eq. (36), ω_e^i is the weight of sample e updated at iteration i, Z_i is a normalization factor (Eq.(37)), α_i is the weight of classifier f_i (Eq. (38)) measured based on the classification error rate ϵ_i (Eq. (39)), x_e represents the feature vector of sample e and y_e represents the ground truth label of sample e.

$$Z_i = \sum_{e=1}^{N} \omega_e^i exp(-\alpha_i y_e h_i(x_e)), \tag{37}$$

$$\alpha_i = \frac{1}{2} \ln(\frac{1-\epsilon_i}{\epsilon_i}), \tag{38}$$

$$\epsilon_i = \frac{\sum_{e=1}^{N} \omega_e^i \cdot I(y_e \neq f_i(x_e))}{\sum_{e=1}^{N} \omega_e^i}. \tag{39}$$

In Eq. (39), $I(\cdot)$ is an indicator function comparing the ground truth label y_e and the output label $f_i(x_e)$ for each sample e.

According to Eq. (36)-(39), it can be derived that classifier f_i trained at iteration i must classify correctly as many as possible those samples misclassified by the ensemble $\bar{f}_{i-1}$ of classifiers trained at the previous iterations, in order to reduce the error rate ϵ_i. Also, due to the case that classifier f_i pays less attention to those samples that were classified correctly by the ensemble $\bar{f}_{i-1}$ of classifiers trained at the previous iterations, some of such samples may be misclassified by classifier f_i. Therefore, the AdaBoost method is considered to be able to create diversity among classifiers explicitly.

In contrast to AdaBoost, the Gradient Boosting method is aimed at training a classifier at iteration i to fit the negative gradient (residual) estimated at iteration $i-1$, based on the error rate of the ensemble $\bar{f}_{i-1}$ of the classifiers trained at the previous iterations. Moreover, any differential loss functions can be used in the setting of gradient boosting, which overcomes the limitations of the AdaBoost method in terms of the selection of loss functions. Based on the principle of the Gradient Boosting method, a popular decision tree ensemble method, referred to as Gradient Boosting Decision Trees (GBDT) [127], has been developed and used in many application areas. In a regression context, as illustrated in Eq. (40), GBDT is essentially aimed at training a decision tree f_i at each iteration i by optimizing θ_i to fit the residual r_{i-1} resulting from the tree ensemble $\bar{f}_{i-1}$ obtained at iteration $i-1$ for error reduction, i.e., the better the decision tree f_i fits the residual r_{i-1}, the larger the error reduction

would be achieved. As pointed out in [101], the error reduction is achieved primarily through the bias reduction although the variance reduction can also be achieved.

$$L(y, \bar{f}_{i-1}(x) + f_i(x;\theta_i)) = (y - \bar{f}_{i-1}(x) - f_i(x;\theta_i))^2 = (r_{i-1} - f_i(x;\theta_i))^2. \quad (40)$$

Based on the Bagging, Random Subspace and Boosting approaches, there have been a variety of decision tree ensemble methods developed by introducing specific ways of diversity creation. In particular, Dynamic Random Forest [124] (a variant of Random Forest) involves training M decision trees on M training sets with different weight distributions over those samples, where the weight ω_e^i of each sample e at each iteration i is set heuristically to be equal to the proportion of base classifiers correctly classifying sample e to the total number of base classifiers obtained so far. Rotation Forest [128] involves using Principal Component Analysis (PCA) [127] over M iterations to draw M transformed feature sets, such that M diverse decision trees are trained. Furthermore, Rotation Random Forest [129] was developed as a variant of Rotation Forest, which involves using PCA or Linear Discriminant Analysis (LDA) [127] to transform each feature subset selected randomly for generating each specific node of a decision tree. Extremely Randomized Trees (Extra-Tree) [130] involves not only randomly selecting a feature subset for generating each specific node of a decision tree but also randomly selecting a numeric value as the cut-point at the tree node if the split attribute is continuous. Random Feature Weights for decision tree ensemble construction [131] was designed to assign each feature a random weight (ranged in [0, 1]) for training a decision tree at each iteration i. In this setting, M different decision trees are trained using M feature sets with different weight distributions over those features. Forest by Penalizing Attributes (Forest PA) [132] was designed to assign each attribute *Attr* a weight heuristically at each iteration i, based on the level of the tree (trained at iteration $i-1$) in which the node (corresponding to attribute *Attr*) is located.

In addition to the above introduced methods, it is also a popular strategy of data input manipulation to create diversity through training multiple classifiers on different sets of features extracted in different ways [133, 134]. In the era of deep learning, a new type of decision tree ensemble referred to as 'Deep Forest' [103, 135] has become more popular. The pilot study was reported in [103], which introduces the gcForest method that aims at producing a cascade of decision forests, i.e., creating an ensemble of ensembles. In particular, the major idea of gcForest is to train a model that involves multiple layers and multiple decision forests (an ensemble of decision tree ensembles) in each layer, where the feature space is dynamically changed every time a new layer is added. In other words, some new features, which are generated as outputs in each layer lr_i, are used as inputs for the next layer lr_{i+1}, where all the original features are kept for each layer. In this context, the ensembles of decision forests in different layers are produced using different sets of features, so those ensembles of decision forests trained in different layers are considered to involve diversity created through diversification of feature sets. Based on gcForest, some variants have been

developed later on through setting different strategies of feature space update, such as multi-layered GBDT [136], Deep Extra-Tree [137], Deep Multigrained Cascade Forest [138], Densely Connected Deep Random Forest [139], Rotation-based deep forest [140], Siamese Deep Forest [141].

In terms of data output manipulation, a popular way is to transform a multi-class classification problem into a number of binary classification problems through binary decomposition. Popular strategies of decomposition include one-vs-one (OVO) [142], one-vs-rest (OVR) [143], many-vs-many (MVM) strategy [142]. In the setting of binary decomposition, an ensemble of binary classifiers is created and the error correcting output codes (ECOC) strategy [144] is commonly used for fusing the outputs of the binary classifiers to finally classify a new sample. Also, ECOC has shown its effectiveness in improving the diversity between binary classifiers in the setting of end-to-end neural network training [145]. More recently, N-nary decomposition has been proposed in [146] as a generalization of binary decomposition. In addition, two other representative ways of output manipulation for diversity enhancement are referred to as 'Output Flipping' and 'Output Smearing', which have been proposed and experimented in [147].

In addition to data manipulation, some other ways can also be taken in practice for diversity creation, which include manipulation of model architectures, differentiation of starting points in the hypothesis space and diversification of learning strategies. The manipulation of model architectures can be applied to decision tree learning, leading to a tree ensemble that contains both binary and multi-way trees, e.g., combining a binary tree trained by CART and two multi-way trees trained by ID3 and C4.5 [148]. Also, in the setting of neural network learning, different types of networks can be produced to form an ensemble through manipulating the network architectures [112]. Differentiation of starting points in the hypothesis space can be applied to neural network learning through random initialization of weights [100] over multiple iterations for training complementary models. In the setting of decision tree learning, starting points in the hypothesis space can be differentiated by selecting different attributes for the root nodes of the trees [149]. In addition, diversification of learning strategies can be achieved by training heterogeneous classifiers through using different learning algorithms [103, 135], or using different hyper-parameter settings of the same learning algorithm, e.g., combination of various loss functions [150].

3. Applications

This section discusses the importance of estimating epistemic uncertainty in several popular applications. These applications include computer vision and natural language processing (NLP). In the following subsections, we first review the applications of epistemic uncertainty learning in computer vision, and then explain how epistemic uncertainty learning has applied to NLP.

3.1. Computer vision

In computer vision, uncertainty is taken into account in variety of applications such as image classification [151, 152], segmentation [83, 153], camera relocalization [154], object detection [155, 156, 157], image/video retrieval (restoration) [158, 159], in the setting of Bayesian and ensemble learning. Image classification and segmentation are among the most popular applications of DL models. The former categorize all objects in an image into a single class, while later aims to assign a label to each pixel in a single image in which pixels from a label share specific properties.

Both classification and segmentation have been widely used for medical image analysis. Although the state-of-the-art supervised learning models can produce precise predictions, they are uncertain about the quality of their predictions. Since the size and shape of diseases are different, and they locate across the patient's body, it is vital to address uncertainties and make predictions interpretable and reliable. Known et al. [153] proposed an uncertainty estimation method using the Bayesian neural networks for stroke lesion segmentation. This method finds a relationship between variance and means of a multi-modal random value. Abdar et al. [151] integrated an ensemble MCD into a multi-model learning framework, which receives chest X-ray (CXR) and computed tomography (CT) images as inputs, to estimate uncertainty in identifying COVID19 cases. Study [160] proposed an uncertainty-aware framework for grading diabetic retinopathy. This framework built a Gaussian sampling approach based on multiple instance learning strategies to infer the grade of images.

Object detection is another popular application of supervised learning models that are being extensively used in autonomous cars. Any mistake in their predictions may cause catastrophic damages or even fatality; therefore, it is vital to estimate the reliability of their predictions. In this regard, prediction surface uncertainty [156], denoted as PURE, was proposed to estimate the predictive uncertainty. This model formulates the object detection task as a regression problem to locate objects in a 2D-camera view image and uses MCD to estimate the uncertainty of the model. Study [161] proposed an uncertainty-aware model for the detection of both salient and camouflaged objects. Specifically, the contradicting attributes of these two tasks were modeled using a similarity measure technique. In addition, an adversarial learning model was proposed to compute the network confidence score.

As the basis of downstream image classification and segmentation tasks, image restoration is an inverse image degradation process. Specifically, it processes the degraded image caused by the imaging device subject to external interference and restores a high-quality image approximating the original image before being degraded. In image restoration tasks, the degraded images are samples with high-level aleatoric uncertainty. Study [158] estimated uncertainty resulting from the undersampled source data. It enhanced the quality of reconstructed images by utilizing a specific network branch to study inherent aleatoric uncertainty arising from noise data. While epistemic uncertainty was superb at estimating the reliability of restored images, satisfying the requirements of safety-critical fields, such as magnetic resonance (MR) images reconstruction. As Begoli et al. [162] concluded that understanding predic-

tion system structure and defensibly quantifying uncertainty is significantly beneficial for medical AI applications. Tanno et al. [163] combined a 3D subpixel-CNN based framework with Bayesian image quality transfer (IQT) [164] to solve diffusion MRI reconstruction problems. They described intrinsic uncertainty as an irreducible variance of mapping low-resolution(LR) to high-resolution(HR) images and defined the degree of ambiguity in the model parameters as parameter uncertainty captured by variational dropout. Subsequently, Schlemper et al. [165] introduced MCD into reconstruction networks, demonstrating the competitive performance of quantifying epistemic uncertainty by utilizing Bayesian methods, especially dealing with test samples which out of training data distribution and superior to overparametrised deterministic networks.

Epistemic uncertainty learning techniques have been applied to other image restoration tasks such as denoising [166, 167], deraining [168]. For instance, Cheng et al. [166] presented an MCMC-based Stochastic gradient Langevin dynamics (SGLD) framework to approximate the posterior distribution to improve performance in image denoising tasks. Serra et al. [167] proposed a fast variational inference framework for solving the sparse representation-related problems in image processing and successfully applied it to the denoising problem.

In addition, several studies have addressed the epistemic uncertainty in analyzing video streams. Huang et al. [169] utilized the similarity of consecutive frames,i.e., temporal property, in videos. They proposed region-based temporal aggregation (RTA) framework, which dramatically speeds up MC-dropout in video segmentation tasks, to estimate uncertainty by calculating the moving average of prediction in consecutive frames to simulate the sampling procedure. Study [170] is the first learning-based solution for the bronchoscopic localization, which estimates uncertainty utilizing VI to conduct video-CT registration.

3.2. Natural language processing

In natural language processing tasks, various metrics of uncertainty quantification have been studied [11, 171, 172, 12, 13] in the context of either Bayesian deep learning or ensemble learning.

In setting of Bayesian deep learning, Xiao and Wang [11] proposed novel methods of quantifying epistemic and aleatoric uncertainties in sentiment analysis, named entity recognition and language modeling tasks, and the experimental results show that learning to quantify uncertainty is not only necessary in measuring the prediction confidence but also useful in improving the model performance. Dong et al. [171] outlined three major causes of uncertainty and designed various metrics for quantifying these factors and estimating confidence scores that indicate the likelihood of correct predictions made by a model. The experimental results reported in [171] show that the proposed confidence model outperforms those methods that rely on confidence scores based on posterior probability, and the interpretation of uncertainty is also improved in comparison with simply using attention scores. Wang et al. [12]

proposed to quantify the epistemic uncertainty for measuring the prediction confidence of a neural machine translation model and their experimental results indicate that the performance of machine translation can be improved significantly through uncertainty-based estimation of prediction confidence.

In the setting of ensemble learning, Shen et al [172] investigated applying Gaussian processes and random forests for measuring the uncertainty in document quality predictions. The experimental results reported in [172] indicate that both Gaussian processes and random forests can be used effectively in predicting the quality of Wikipedia articles alongside an estimate of the uncertainty concerning the inconsistent outputs of various models. He et al. [13] proposed to improve the confidence of winning score for generating accurate uncertainty score. In particular, a model, which consists of three parts, namely, "mix-up", "self-ensembling" and "distinctiveness score", is proposed in the setting of deep neural networks for reducing the impact of the overconfidence of winning score and also taking into account the impacts of other types of uncertainty. The experimental results reported in [13] indicate that accurate scores of uncertainty can be obtained using the proposed model and the performance of text classification can be improved by assigning those uncertain predictions to domain experts.

4. Discussions and Conclusions

In this survey, we provided a hierarchical categorization of the epistemic (model) uncertainty learning methods, i.e., Bayesian and ensemble methods. Bayesian methods formulate epistemic uncertainty as a posterior distribution over the weight parameters. Since these methods need to compute posterior, they cannot perform inference analytically but can be approximated. In this regard, we discuss four widely used approximation techniques, including variational inference (VI), Monte Carlo dropout (MCD), Markov Chain Monte Carlo (MCMC), and Laplace approximation. Each of these techniques has several advantages and disadvantages [36]. Among them, MCD techniques are easy to implement and don't need to change the training process. However, they are not reliable for out-of-distribution samples and require multiple sampling when performing inference. VI techniques benefit from stochastic optimization methods and are suitable for big data sets. However, they are computationally complex.MCMC techniques can approximate exact posterior, but they are very slow and fail to converge. Although the Laplace approximation techniques have a simple procedure, they perform poorly due to ignoring the global properties of the real posterior.

In contrast, ensemble methods formulate epistemic uncertainty as the variance of the outputs of base models. The epistemic uncertainty is also referred to as ensemble diversity, which is considered as a key factor of successful ensemble learning. In particular, existing works have illustrated mathematically how the ensemble diversity impacts the generalization performance in the context of bias-variance decomposition. There have been quite a lot of studies conducted for diversity quantification

and creation, which have provided useful guidance on how to construct effectively a high quality ensemble leading to the improvement of the generalization performance. However, there is still not yet a formally accepted definition of the term 'diversity' [96, 101, 135], which indicates that different measures of diversity were designed from different views of diversity and the ensemble diversity was usually created in different heuristic ways [103]. Moreover, a great number of studies have been conducted towards optimizing the ensemble accuracy and the diversity simultaneously and some works also involve introducing new metrics of diversity quantification towards enhancing heuristically the relationship between the ensemble accuracy and the diversity. However, the above-mentioned relationship still needs to be explored further in depth to make it more clear how the simultaneous optimization of the ensemble accuracy and the diversity can be achieved more effectively.

4.1. Research Gaps

Despite considerable progress in handling the epistemic uncertainty in supervised learning models, there exist several challenging issues that must be addressed in the future. We found several research gaps that need further investigations, as follows:

- **Methodology**: Although supervised learning approaches have been widely applied to solve computer vision and NLP problems, most of the existing studies fail to quantify uncertainty in practice. They usually use ideal (standard) data sets and inject a uniform random noise to evaluate their performance, which is unrealistic in real-world problems. In practice, the performance of learning from data sets are affected by uncertain distributions; therefore, it is crucial to develop robust techniques for learning uncertainty. Moreover, in NLP, the uncertainty on the contexts of words is naturally present in the text due to the insufficient amount of data but very few studies on epistemic uncertainty have been conducted in this aspect, which indicates the necessity of further studies on uncertainty in text processing. In addition, most of the studies estimated the uncertainty in supervised learning models, while little attention has been paid to other learning strategies such as semi-supervised learning [173], multi-modal learning [174], reinforcement learning [175], active learning [176], transfer learning [177], graph learning [178], etc. From the perspective of algorithm optimization, choosing suitable epistemic uncertainty quantification methods according to specific tasks and algorithm characteristics, and generating an optimized learning strategy based on quantified uncertainty, is worth further exploration. It may be an effective way to improve the performance of deep neural networks with different structural characteristics and other classic learning algorithms. For example, there are several excellent evolutionary computation algorithms, such as particle swarm optimization [179, 180, 181], that have received extensive attention from researchers in the post-deep learning era. However, there is little research work on quantifying the uncertainty of such algorithms. We believe

that the epistemic uncertainty learning techniques can be used to improve the stability of the optimization process.

- **Lack of data set**: For the topic of model uncertainty quantification, there are not yet benchmark databases designed particularly. The data sets used in this study are from areas of CV and NLP. Nonetheless, it is one of the basements for studying epistemic uncertainty quantification. Analyzing epistemic uncertainty based on bias-variance decomposition may be a breakthrough in constructing a benchmark data set that reflects the effectiveness of epistemic uncertainty quantification techniques fairly. We will explore this further in our future work.

- **Lack of standard evaluation protocol**: Existing uncertainty learning techniques are being evaluated based on measurable quantities such as performance on out-of-distribution detection. However, the details of such performance evaluation strategy may vary for different studies, which leads to an unfair comparison among various techniques. Therefore, it is vital to have a standard protocol for evaluating the effectiveness of uncertainty quantification techniques directly. Based on the bias and variance decomposition, which is mentioned in this work, making theoretical exploration of the quality of the uncertainty estimation is also a good future research direction.

- **Availability of data and code:** This can help researchers to reproduce results, enhance their performance and conduct a fair comparison. However, the majority of studies do not make the relevant codes and data available.

4.2. Concluding remarks

Enabling supervised learning models to quantify their uncertainty is vital for many real-world applications such as safety-related problems. This survey first explained the importance of addressing the epistemic uncertainty in supervised learning models and discussed it in terms of bias and variance. Then, we reviewed the epistemic uncertainty learning techniques in supervised learning over the last five years. We provided a hierarchical categorization of these techniques and introduced the representative models of each category along with their applications. Specifically, we discussed two widely used epistemic uncertainty learning techniques, i.e., Bayesian approximation and ensemble learning. In addition, several research gaps have been pointed out as potential future research directions. It is aimed to promote the concept of epistemic uncertainty learning.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (Grants 61976141, 62176160 and 61732011), in part by National Key R&D

Program of China (Grant 2021YFE0203700), in part by the Natural Science Foundation of Shenzhen (University Stability Support Program no. 20200804193857002), and in part by the Interdisciplinary Innovation Team of Shenzhen University.

References

[1] X. Wang, Y. Zhao, F. Pourpanah, Recent advances in deep learning, International Journal of Machine Learning and Cybernetics 11 (2020) 747–750.

[2] F. Pourpanah, M. Abdar, Y. Luo, X. Zhou, R. Wang, C. P. Lim, X.-Z. Wang, A review of generalized zero-shot learning methods, arXiv:2011.08641.

[3] S. Rezvani, X. Wang, F. Pourpanah, Intuitionistic fuzzy twin support vector machines, IEEE Transactions on Fuzzy Systems 27 (11) (2019) 2140–2151.

[4] Y. Luo, X. Wang, F. Pourpanah, Dual vaegan: A generative model for generalized zero-shot learning, Applied Soft Computing 107 (2021) 107352.

[5] S. Garg, S. P. Awate, Perfect mcmc sampling in bayesian mrfs for uncertainty estimation in segmentation, in: Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, 2018, pp. 673–681.

[6] T. Nair, D. Precup, D. L. Arnold, T. Arbel, Exploring uncertainty measures in deep networks for multiple sclerosis lesion detection and segmentation, Medical image analysis 59 (2020) 101557.

[7] J. Wang, Z. He, S. Huang, H. Chen, W. Wang, F. Pourpanah, Fuzzy measure with regularization for gene selection and cancer prediction, International Journal of Machine Learning and Cybernetics 12 (8) (2021) 2389–2405.

[8] F. Pourpanah, B. Zhang, R. Ma, Q. Hao, Anomaly detection and condition monitoring of uav motors and propellers, in: IEEE SENSORS, 2018, pp. 1–4.

[9] Y. He, C. Zhu, J. Wang, M. Savvides, X. Zhang, Bounding box regression with uncertainty for accurate object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2888–2897.

[10] F. Kraus, K. Dietmayer, Uncertainty estimation in one-stage object detection, in: Proceedings of the IEEE Intelligent Transportation Systems Conference, 2019, pp. 53–60.

[11] Y. Xiao, W. Y. Wang, Quantifying uncertainties in natural language processing tasks, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, pp. 7322–7329.

[12] S. Wang, Y. Liu, C. Wang, H. Luan, M. Sun, Improving back-translation with uncertainty-based confidence estimation, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing and the International Joint Conference on Natural Language Processing, 2019, pp. 791–802.

[13] J. He, X. Zhang, S. Lei, Z. Chen, F. Chen, A. Alhamadani, B. Xiao, C.-T. Lu, Towards more accurate uncertainty estimation in text classification, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2020, pp. 8362–8372.

[14] S. Däubener, L. Schönherr, A. Fischer, D. Kolossa, Detecting adversarial examples for speech recognition via uncertainty quantification, arXiv preprint arXiv:2005.14611.

[15] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: A deep convolutional encoder-decoder architecture for image segmentation, IEEE transactions on pattern analysis and machine intelligence 39 (12) (2017) 2481–2495.

[16] N. Zeng, H. Li, Z. Wang, W. Liu, S. Liu, F. E. Alsaadi, X. Liu, Deep-reinforcement-learning-based images segmentation for quantitative analysis of gold immunochromatographic strip, Neurocomputing 425 (2021) 173–180.

[17] N. Zeng, Z. Wang, H. Zhang, K.-E. Kim, Y. Li, X. Liu, An improved particle filter with a novel hybrid proposal distribution for quantitative analysis of gold immunochromatographic strips, IEEE Transactions on Nanotechnology 18 (2019) 819–829.

[18] F. Fang, J. Li, T. Zeng, Soft-edge assisted network for single image super-resolution, IEEE Transactions on Image Processing 29 (2020) 4656–4668.

[19] W. Liu, Z. Wang, L. Tian, S. Lauria, X. Liu, Melt pool segmentation for additive manufacturing: A generative adversarial network approach, Computers & Electrical Engineering 92 (2021) 107183.

[20] E. Hullermeier, W. Waegeman, Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods, Machine Learning 110 (3) (2021) 457–506.

[21] X. Wang, Y. He, Learning from uncertainty for big data: future analytical challenges and strategies, IEEE Systems, Man, and Cybernetics Magazine 2 (2) (2016) 26–31.

[22] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. Dillon, B. Lakshminarayanan, J. Snoek, Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift, Advances in neural information processing systems 32 (2019) 1–25.

[23] M. S. Ayhan, P. Berens, Test-time data augmentation for estimation of heteroscedastic aleatoric uncertainty in deep neural networks, in: International conference on Medical Imaging with Deep Learning, 2018, pp. 1–9.

[24] A. G. Roy, S. Conjeti, N. Navab, C. Wachinger, Bayesian quicknat: Model uncertainty in deep whole-brain segmentation for structure-wise quality control, NeuroImage 195 (2019) 11–22.

[25] H. Shen, S. Chen, R. Wang, A study on the uncertainty of convolutional layers in deep neural networks, International Journal of Machine Learning and Cybernetics 12 (6) (2021) 1853–1865.

[26] A. Kendall, Y. Gal, What uncertainties do we need in bayesian deep learning for computer vision?, Advances in Neural Information Processing Systems 30 (2017) 5574–5584.

[27] X. Wang, H. Xing, Y. Li, Q. Hua, C. Dong, W. Pedrycz, A study on relationship between generalization abilities and fuzziness of base classifiers in ensemble learning, IEEE Transactions on Fuzzy Systems 23 (5) (2014) 1638–1654.

[28] X. Wang, R. Wang, C. Xu, Discovering the relationship between generalization and uncertainty by incorporating complexity of classification, IEEE transactions on cybernetics 48 (2) (2017) 703–715.

[29] X. Zhou, X. Wang, C. Hu, R. Wang, An analysis on the relationship between uncertainty and misclassification rate of classifiers, Information Sciences 535 (2020) 16–27.

[30] J. Gao, Bias-variance decomposition of absolute errors for diagnosing regression models of continuous data, Patterns 2 (8) (2021) 100309.

[31] J. Friedman, T. Hastie, R. Tibshirani, et al., The elements of statistical learning, Vol. 1, Springer series in statistics New York, 2001.

[32] H. D. Kabir, A. Khosravi, M. A. Hosen, S. Nahavandi, Neural network-based uncertainty quantification: A survey of methodologies and applications, IEEE access 6 (2018) 36218–36234.

[33] R. H. Hariri, E. M. Fredericks, K. M. Bowers, Uncertainty in big data analytics: survey, opportunities, and challenges, Journal of Big Data 6 (1) (2019) 1–16.

[34] H. Wang, D.-Y. Yeung, A survey on bayesian deep learning, ACM Computing Surveys (CSUR) 53 (5) (2020) 1–37.

[35] L. V. Jospin, W. Buntine, F. Boussaid, H. Laga, M. Bennamoun, Hands-on bayesian neural networks-a tutorial for deep learning users, ACM Comput. Surv 1 (1).

[36] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U. R. Acharya, et al., A review of uncertainty quantification in deep learning: Techniques, applications and challenges, Information Fusion.

[37] J. Gawlikowski, C. R. N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher, et al., A survey of uncertainty in deep neural networks, arXiv preprint arXiv:2107.03342.

[38] Y. Gal, Uncertainty in deep learning, Ph.D. thesis, Cambridge University, Cambridge, UK (2016).

[39] Y. Anzai, Pattern recognition and machine learning, Elsevier, 2012.

[40] G. E. Hinton, D. van Camp, Keeping the neural networks simple by minimizing the description length of the weights, in: Proceedings of the Sixth Annual Conference on Computational Learning Theory, 1993, p. 5–13.

[41] S. Kullback, R. A. Leibler, On information and sufficiency, The Annals of Mathematical Statistics 22 (1) (1951) 79 – 86.

[42] A. Graves, Practical variational inference for neural networks, Advances in Neural Information Processing Systems 24.

[43] C. Blundell, J. Cornebise, K. Kavukcuoglu, D. Wierstra, Weight uncertainty in neural network, in: Proceedings of the International Conference on Machine Learning, 2015, pp. 1613–1622.

[44] D. P. Kingma, M. Welling, Auto-encoding variational bayes, arXiv preprint arXiv:1312.6114.

[45] D. P. Kingma, T. Salimans, M. Welling, Variational dropout and the local reparameterization trick, Advances in neural information processing systems 28 (2015) 1–9.

[46] D. Rezende, S. Mohamed, Variational inference with normalizing flows, in: Proceedings of the International conference on Machine Learning, 2015, pp. 1530–1538.

[47] J. Zeng, A. Lesnikowski, J. M. Alvarez, The relevance of bayesian layer positioning to model uncertainty in deep bayesian active learning, Advances in neural information processing systems (2018) 1–6.

[48] G. Zhang, S. Sun, D. Duvenaud, R. Grosse, Noisy natural gradient as variational inference, in: Proceedings of the International Conference on Machine Learning, 2018, pp. 5852–5861.

[49] S. I. Amari, Neural learning in structured parameter spaces - natural riemannian gradient, Advances in neural information processing systems 9 (1997) 127–133.

[50] K. Osawa, S. Swaroop, A. Jain, R. Eschenhagen, R. E. Turner, R. Yokota, M. E. Khan, Practical deep learning with bayesian principles, in: Proceedings of the International Conference on Neural Information Processing Systems, 2019, pp. 4287–4299.

[51] M. Khan, D. Nielsen, V. Tangkaratt, W. Lin, Y. Gal, A. Srivastava, Fast and scalable bayesian deep learning by weight-perturbation in adam, in: International Conference on Machine Learning, 2018, pp. 2611–2620.

[52] A. Mishkin, F. Kunstner, D. Nielsen, M. Schmidt, M. E. Khan, Slang: fast structured covariance approximations for bayesian deep learning with natural gradient, in: Proceedings of the International Conference on Neural Information Processing Systems, 2018, pp. 6248–6258.

[53] J. Heo, H. B. Lee, S. Kim, J. Lee, K. J. Kim, E. Yang, S. J. Hwang, Uncertainty-aware attention for reliable interpretation and prediction, Advances in Neural Information Processing Systems 31 (2018) 909–918.

[54] R. M. Neal, Bayesian learning via stochastic dynamics, Advances in neural information processing systems (1993) 475–482.

[55] Y. Gal, Z. Ghahramani, Dropout as a bayesian approximation: Representing model uncertainty in deep learning, in: Proceedings of the International Conference on Machine Learning, 2016, pp. 1050–1059.

[56] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, Journal of Machine Learning Research 15 (56) (2014) 1929–1958.

[57] Y. Gal, J. Hron, A. Kendall, Concrete dropout, Advances in neural information processing systems.

[58] J. Mukhoti, Y. Gal, Evaluating bayesian deep learning methods for semantic segmentation, arXiv:1811.12709.

[59] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European Conference on Computer Vision, 2018, pp. 801–818.

[60] K. Brach, B. Sick, O. Dürr, Single shot mc dropout approximation, arXiv:2007.03293.

[61] N. Kennamer, A. T. Ihler, D. Kirkby, Empirical study of mc-dropout in various astronomical observing conditions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2019, pp. 17–20.

[62] M. Abdar, M. Samami, S. Dehghani Mahmoodabad, T. Doan, B. Mazoure, R. Hashemifesharaki, L. Liu, A. Khosravi, U. R. Acharya, V. Makarenkov, S. Nahavandi, Uncertainty quantification in skin cancer classification using three-way decision-based bayesian deep learning, Computers in Biology and Medicine (2021) 104418.

[63] M. Abdar, M. A. Fahami, S. Chakrabarti, A. Khosravi, P. Pławiak, U. R. Acharya, R. Tadeusiewicz, S. Nahavandi, Barf: A new direct and cross-based binary residual feature fusion with uncertainty-aware module for medical image classification, Information Sciences 577 (2021) 353–378.

[64] G. Wang, W. Li, M. Aertsen, J. Deprest, S. Ourselin, T. Vercauteren, Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks, Neurocomputing 338 (2019) 34–45.

[65] A. Loquercio, M. Segu, D. Scaramuzza, A general framework for uncertainty estimation in deep learning, IEEE Robotics and Automation Letters 5 (2) (2020) 3153–3160.

[66] L. Bertoni, S. Kreiss, A. Alahi, MonoLoco: Monocular 3d pedestrian localization and uncertainty estimation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 6861–6871.

[67] L. Zhu, N. Laptev, Deep and confident prediction for time series at uber, in: Proceedings of the IEEE International Conference on Data Mining Workshops, 2017, pp. 103–110.

[68] T. Chen, E. Fox, C. Guestrin, Stochastic gradient hamiltonian monte carlo, in: Proceedings of the International Conference on Machine Learning, 2014, pp. 1683–1691.

[69] S. Duane, A. Kennedy, B. J. Pendleton, D. Roweth, Hybrid monte carlo, Physics Letters B 195 (2) (1987) 216–222.

[70] M. Welling, Y. W. Teh, Bayesian learning via stochastic gradient langevin dynamics, in: Proceedings of the International Conference on Machine Learning, 2011, pp. 681–688.

[71] M. Teye, H. Azizpour, K. Smith, Bayesian uncertainty estimation for batch normalized deep networks, in: Proceedings of the International Conference on Machine Learning, 2018, pp. 4907–4916.

[72] R. Chandra, A. Bhagat, M. Maharana, P. N. Krivitsky, Bayesian graph convolutional neural networks via tempered mcmc, arXiv preprint arXiv:2104.08438.

[73] S. Mandt, M. D. Hoffman, D. M. Blei, Stochastic gradient descent as approximate bayesian inference, Journal of Machine Learning Research 18 (2017) 1–35.

[74] R. Zhang, C. Li, J. Zhang, C. Chen, A. G. Wilson, Cyclical stochastic gradient mcmc for bayesian deep learning, in: Proceedings of the International Conference on Learning Representations, 2020, pp. 1–27.

[75] R. Luo, J. Wang, Y. Yang, J. WANG, Z. Zhu, Thermostat-assisted continuously-tempered hamiltonian monte carlo for bayesian learning, Advances in Neural Information Processing Systems 31 (2018) 10673–10682.

[76] W. G. Hoover, Canonical dynamics: Equilibrium phase-space distributions, Physical review A 31 (3) (1985) 1695.

[77] W. J. Maddox, P. Izmailov, T. Garipov, D. P. Vetrov, A. G. Wilson, A simple baseline for bayesian uncertainty in deep learning, Advances in Neural Information Processing Systems 32 (2019) 13153–13164.

[78] P. Izmailov, D. Podoprikhin, T. Garipov, D. Vetrov, A. G. Wilson, Averaging weights leads to wider optima and better generalization, in: Proceedings of the Conference on Uncertainty in Artificial Intelligence, 2018, pp. 876–885.

[79] J. G. Propp, D. B. Wilson, Exact sampling with coupled markov chains and applications to statistical mechanics, Random Structures & Algorithms 9 (1-2) (1996) 223–252.

[80] M. Huber, Perfect sampling using bounding chains, The Annals of Applied Probability 14 (2).

[81] S. Hernández, D. Vergara, M. Valdenegro-Toro, F. Jorquera, Improving predictive uncertainty estimation using dropout–hamiltonian monte carlo, Soft Computing 24 (6) (2020) 4307–4322.

[82] E. Akkoyun, S. T. Kwon, A. C. Acar, W. Lee, S. Baek, Predicting abdominal aortic aneurysm growth using patient-oriented growth models with two-step bayesian inference, Computers in biology and medicine 117 (2020) 103620.

[83] X. Cai, M. Pereyra, J. D. McEwen, Uncertainty quantification for radio interferometric imaging–i. proximal mcmc methods, Monthly Notices of the Royal Astronomical Society 480 (3) (2018) 4154–4169.

[84] D. J. MacKay, A practical bayesian framework for backpropagation networks, Neural computation 4 (3) (1992) 448–472.

[85] Z. Hong, Bayesian estimation of stochastic volatility models by integrated nested laplace approximation method, Master's thesis, Shandong University (2019).

[86] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, et al., Overcoming catastrophic forgetting in neural networks, Proceedings of the national academy of sciences 114 (13) (2017) 3521–3526.

[87] H. Ritter, A. Botev, D. Barber, A scalable laplace approximation for neural networks, in: Proceedings of the International Conference on Learning Representations, 2018, pp. 1–15.

[88] H. Ritter, A. Botev, D. Barber, Online structured laplace approximations for overcoming catastrophic forgetting, in: Proceedings of the International Conference on Neural Information Processing Systems, 2018, pp. 3742–3752.

[89] J. Lee, M. Humt, J. Feng, R. Triebel, Estimating model uncertainty of neural networks in sparse information form, in: Proceedings of the International Conference on Machine Learning, PMLR, 2020, pp. 5702–5713.

[90] C. Schillings, B. Sprungk, P. Wacker, On the convergence of the laplace approximation and noise-level-robustness of laplace-based monte carlo methods for bayesian inverse problems, Numerische Mathematik 145 (4) (2020) 915–971.

[91] C. Margossian, A. Vehtari, D. Simpson, R. Agrawal, Hamiltonian monte carlo using an adjoint-differentiated laplace approximation: Bayesian inference for latent gaussian models and beyond, in: Proceedings of the IEEE Conference on Neural Information Processing Systems;, 2020, pp. 1–18.

[92] E. Daxberger, E. Nalisnick, J. U. Allingham, J. Antorán, J. M. Hernández-Lobato, Bayesian deep learning via subnetwork inference, in: Proceedings of the International Conference on Machine Learning, PMLR, 2021, pp. 2510–2521.

[93] C. S. Perone, R. P. Silveira, T. Paula, L2m: Practical posterior laplace approximation with optimization-driven second moment estimation, arXiv preprint arXiv:2107.04695.

[94] B. Lakshminarayanan, A. Pritzel, C. Blundell, Simple and scalable predictive uncertainty estimation using deep ensembles, in: Proceedings of the Conference on Neural Information Processing Systems, 2017, pp. 1–15.

[95] L. Tran, B. S. Veeling, K. Roth, J. Swiatkowski, J. V. Dillon, S. Mandt, J. Snoek, T. Salimans, S. Nowozin, R. Jenatton, Hydra: Preserving ensemble

diversity for model distillation, in: Proceedings of the International Conference on Machine Learning Workshop, 2020, pp. 1–10.

[96] L. I. Kuncheva, C. J. Whitaker, Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy, Machine Learning 51 (2) (2003) 181–207.

[97] G. D. C. Cavalcanti, L. S. Oliveira, T. J. M. Moura, G. V. Carvalho, Combining diversity measures for ensemble pruning, Pattern Recognition Letters 74 (apr.15) (2016) 38–45.

[98] A. Krogh, J. Vedelsby, Neural network ensembles, cross validation and active learning, in: Proceedings of the International Conference on Neural Information Processing Systems, 1994, p. 231–238.

[99] S. Geman, E. Bienenstock, R. Doursat, Neural networks and the bias/variance dilemma, Neural Computation 4 (1) (1992) 1–58.

[100] G. Brown, J. Wyatt, R. Harris, X. Yao, Diversity creation methods: a survey and categorisation, Information Fusion 6 (1) (2005) 5–20.

[101] Z.-H. Zhou, Ensemble Methods: Foundations and Algorithms, Chapman and Hall/CRC, London, 2012.

[102] G. Brown, L. I. Kuncheva, Good and bad diversity in majority vote ensembles, in: Proceedings of the International Workshop on Multiple Classifier Systems, 2010, pp. 124–133.

[103] Z.-H. Zhou, J. Feng, Deep forest: Towards an alternative to deep neural networks, in: Proceedings of the International Joint Conference on Artificial Intelligence, 2017, pp. 3553–3559.

[104] K. Tumer, J. Ghosh, Error correlation and error reduction in ensemble classifiers, Connection Science 8 (3-4) (1996) 385–403.

[105] K. Tumer, J. Ghosh, Analysis of decision boundaries in linearly combined neural classifiers, Pattern Recognition 29 (2) (1996) 341–348.

[106] A. Sharkey, N. E. Sharkey, Combining diverse neural nets, The Knowledge Engineering Review 12 (3) (1998) 231–247.

[107] W. Krzanowski, D. Partridge, Software diversity: Practical statistics for its measurement and exploitation, Information and Software Technology 39 (10) (1997) 707–717.

[108] X.-C. Yin, K. Huang, C. Yang, H.-W. Hao, Convex ensemble learning with sparsity and diversity, Information Fusion 20 (2014) 49–59.

[109] M. Ahmed, L. Didaci, B. Lavi, G. Fumera, Using diversity for classifier ensemble pruning: An empirical investigation, Theoretical and Applied Informatics 29 (1& 2) (2018) 25–39.

[110] Q. Dai, Y. Rui, Z. Liu, Considering diversity and accuracy simultaneously for ensemble pruning, Applied Soft Computing 58 (2017) 75–91.

[111] N. Dvornik, J. Mairal, C. Schmid, Diversity with cooperation: Ensemble methods for few-shot classification, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 3722–3730.

[112] S. Zhang, M. Liu, J. Yan, The diversified ensemble neural network, Advances in Neural Information Processing Systems 33 (2020) 1–11.

[113] Y. Bian, H. Chen, When does diversity help generalization in classification ensembles?, IEEE Transactions on Cybernetics (2021) 1–17.

[114] Y. Wu, L. Liu, Z. Xie, K.-H. Chow, W. Wei, Boosting ensemble accuracy by revisiting ensemble diversity metrics, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 16469–16477.

[115] L. Breiman, Bagging predictors, Machine Learning 24 (2) (1996) 123–140.

[116] T. K. Ho, The random subspace method for constructing decision forests, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (8) (1998) 832–844.

[117] Y. Freund, Boosting a weak learning algorithm by majority, Information and Computation 121 (2) (1995) 256–285.

[118] E. Bauer, R. Kohavi, An empirical comparison of voting classification algorithms: Bagging, boosting, and variants, Machine Learning 36 (1999) 105–139.

[119] K. M. Ting, I. H. Witten, Stacking bagged and dagged models, in: Proceedings of the International Conference on Machine Learning, 1997, pp. 367–375.

[120] K. M. Ting, B. T. Low, Model combination in the multiple-data-batches scenario, in: Proceedings of European Conference on Machine Learning, 1997, pp. 250–265.

[121] R. Brylla;*, R. Gutierrez-Osunab, F. Queka, Attribute bagging: improving accuracy of classifier ensembles by using random feature subsets, Pattern Recognition 36 (6) (2003) 1291–1302.

[122] K. Fawagreh, M. M. Gaber, E. Elyan, Random forests: from early developments to recent advancements, Systems Science & Control Engineering An Open Access Journal 2 (1) (2014) 602–609.

[123] L. Breiman, Random forests, Machine Learning 45 (1) (2001) 5–32.

[124] S. Bernard, S. Adam, L. Heutte, Dynamic random forests, Pattern Recognition Letters 33 (12) (2012) 1580–1586.

[125] Y. Freund, R. E. Schapire, Experiments with a new boosting algorithm, in: Proceedings of the International Conference on Machine Learning:, 1996, pp. 148–156.

[126] J. H. Friedman, Greedy function approximation: A gradient boosting machine, The Annals of Statistics 29 (5) (2001) 1189–1232.

[127] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer, London, 2009.

[128] J. J. Rodriguez, L. I. Kuncheva, C. J. Alonso, Rotation forest: A new classifier ensemble method, IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (10) (2006) 1619–1630.

[129] L. Zhang, P. N. Suganthan, Random forests with ensemble of feature spaces, Pattern Recognition 47 (10) (2014) 3429–3437.

[130] P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees, Machine Learning 63 (1) (2006) 3–42.

[131] J. Maudes, J. J. Rodriguez, C. Garcia-Osorio, N. Garcia-Pedrajas, Random feature weights for decision tree ensemble construction, Information Fusion 13 (1) (2012) 20–30.

[132] M. N. Adnan, M. Z. Islam, Forest pa: Constructing a decision forest by penalizing attributes used in previous trees, Expert Systems With Applications 89 (2017) 389–403.

[133] H. Gunes, M. Piccardi, Affect recognition from face and body: Early fusion vs. late fusion, in: IEEE International Conference on Systems, Man and Cybernetics, 2005, pp. 3437–3443.

[134] T. Baltrusaitis, C. Ahuja, L. P. Morency, Multimodal machine learning: A survey and taxonomy, IEEE Transactions on Pattern Analysis and Machine Intelligence PP (99) (2017) 1–1.

[135] Z. Zhou, J. Feng, Deep forest, National Science Review 6 (1) (2019) 74–86.

[136] J. Feng, Y. Yu, Z.-H. Zhou, Multi-layered gradient boosting decision trees, in: Proceedings of the Conference on Neural Information Processing Systems, 2018, pp. 1–16.

[137] A. Berrouachedi, R. Jaziri, G. Bernard, Deep extremely randomized trees, in: Proceedings of the International Conference on Neural Information Processing, 2019, pp. 717–729.

[138] X. Liu, R. Wang, Z. Cai, Y. Cai, X. Yin, Deep multigrained cascade forest for hyperspectral image classification, IEEE Transactions on Geoscience and Remote Sensing 57 (10) (2019) 8169–8183.

[139] X. Cao, R. Li, Y. Ge, B. Wu, L. Jiao, Densely connected deep random forest for hyperspectral imagery classification, International journal of remote sensing 40 (9-10) (2019) 3606–3621.

[140] X. Cao, L. Wen, Y. Ge, J. Zhao, L. Jiao, Rotation-based deep forest for hyperspectral imagery classification, IEEE Geoscience and Remote Sensing Letters 16 (7) (2019) 1105–1109.

[141] L. V. Utkin, M. A. Ryabinin, A siamese deep forest, Knowledge-Based Systems 139 (jan.1) (2018) 13–22.

[142] P. A. Gutiérrez, M. Pérez-Ortiz, J. Sánchez-Monedero, F. Fernández-Navarro, C. Hervás-Martínez, Ordinal regression methods: Survey and experimental study, IEEE Transactions on Knowledge and Data Engineering 28 (1) (2016) 127–146.

[143] M. Wozniak, M. Grana, E. Corchado, A survey of multiple classifier systems as hybrid systems, Information Fusion 16 (2014) 3–17.

[144] T. G. Dietterich, G. Bakiri, Solving multiclass learning problems via error-correcting output codes, Journal of Artificial Intelligence Research 2 (1) (1994) 263–286.

[145] Y. Song, Q. Kang, W. P. Tay, Error-correcting output codes with ensemble diversity for robust learning in neural networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2021, pp. 9722–9729.

[146] J. T. Zhou, I. W. Tsang, S.-S. Ho, K.-R. Muller, N-ary decomposition for multi-class classification, Machine Learning 108 (2019) 809–830.

[147] L. Breiman, Randomizing outputs to increase prediction accuracy, Machine Learning 40 (2000) 229–242.

[148] S. Bashir, U. Qamar, F. H. Khan, M. Y. Javed, An efficient rule-based classification of diabetes using id3, c4.5, & cart ensembles, in: Proceedings of the International Conference on Frontiers of Information Technology, 2015, pp. 226–231.

[149] M. N. Adnan, M. Z. Islam, Forest cern: A new decision forest building technique, in: Advances in Knowledge Discovery and Data Mining: 20th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Auckland, New Zealand, 2016, pp. 304–315.

[150] H. Hajiabadi, D. Molla-Aliod, R. Monsef, H. S. Yazdi, Combination of loss functions for deep text classifcation, International Journal of Machine Learning and Cybernetics 11 (2020) 751–761.

[151] M. Abdar, S. Salari, S. Qahremani, H.-K. Lam, F. Karray, S. Hussain, A. Khosravi, U. R. Acharya, S. Nahavandi, Uncertaintyfusenet: Robust uncertainty-aware hierarchical feature fusion with ensemble monte carlo dropout for covid-19 detection, arXiv:2105.08590.

[152] Z. Senousy, M. Abdelsamea, M. M. Gaber, M. Abdar, R. U. Acharya, A. Khosravi, S. Nahavandi, Mcua: Multi-level context and uncertainty aware dynamic deep ensemble for breast cancer histology image classification, IEEE Transactions on Biomedical Engineering (2021) 1–1.

[153] Y. Kwon, J.-H. Won, B. J. Kim, M. C. Paik, Uncertainty quantification using bayesian neural networks in classification: Application to biomedical image segmentation, Computational Statistics & Data Analysis 142 (2020) 106816.

[154] A. Kendall, R. Cipolla, Modelling uncertainty in deep learning for camera relocalization, in: Proceedings of the IEEE International Conference on Robotics and Automation, 2016, pp. 4762–4769.

[155] M. Schubert, K. Kahl, M. Rottmann, Metadetect: Uncertainty quantification and prediction quality estimates for object detection, arXiv preprint arXiv:2010.01695.

[156] F. O. Catak, T. Yue, S. Ali, Prediction surface uncertainty quantification in object detection models for autonomous driving, arXiv preprint arXiv:2107.04991.

[157] Y. Chen, K. Xu, D. He, X. Ban, Generating robust real-time object detector with uncertainty via virtual adversarial training, International Journal of Machine Learning and Cybernetics (2021) 1–15.

[158] Z. Zhang, A. Romero, M. J. Muckley, P. Vincent, L. Yang, M. Drozdzal, Reducing uncertainty in undersampled mri reconstruction with active acquisition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2049–2058.

[159] G. Dorta, S. Vicente, L. Agapito, N. D. Campbell, I. Simpson, Structured uncertainty prediction networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5477–5485.

[160] T. Araújo, G. Aresta, L. Mendonça, S. Penas, C. Maia, Ângela Carneiro, A. M. Mendonça, A. Campilho, Dr—graduate: Uncertainty-aware deep learning-based diabetic retinopathy grading in eye fundus images, Medical Image Analysis 63 (2020) 101715.

[161] A. Li, J. Zhang, Y. Lv, B. Liu, T. Zhang, Y. Dai, Uncertainty-aware joint salient object and camouflaged object detection, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 10071–10081.

[162] E. Begoli, T. Bhattacharya, D. Kusnezov, The need for uncertainty quantification in machine-assisted medical decision making, Nature Machine Intelligence 1 (1) (2019) 20–23.

[163] R. Tanno, D. E. Worrall, A. Ghosh, E. Kaden, S. N. Sotiropoulos, A. Criminisi, D. C. Alexander, Bayesian image quality transfer with cnns: exploring uncertainty in dmri super-resolution, in: Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, 2017, pp. 611–619.

[164] R. Tanno, A. Ghosh, F. Grussu, E. Kaden, A. Criminisi, D. C. Alexander, Bayesian image quality transfer, in: Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, 2016, pp. 265–273.

[165] J. Schlemper, D. C. Castro, W. Bai, C. Qin, O. Oktay, J. Duan, A. N. Price, J. Hajnal, D. Rueckert, Bayesian deep learning for accelerated mr image reconstruction, in: Proceedings of the International Workshop on Machine Learning for Medical Image Reconstruction, 2018, pp. 64–71.

[166] Z. Cheng, M. Gadelha, S. Maji, D. Sheldon, A bayesian perspective on the deep image prior, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 5443–5451.

[167] J. G. Serra, M. Testa, R. Molina, A. K. Katsaggelos, Bayesian k-svd using fast variational inference, IEEE Transactions on Image Processing 26 (7) (2017) 3344–3359.

[168] C. Chen, H. Li, Robust representation learning with feedback for single image deraining, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 7742–7751.

[169] P. Huang, W. Hsu, C. Chiu, T. Wu, M. Sun, Efficient uncertainty estimation for semantic segmentation in videos, in: Proceedings of the European Conference on Computer Vision, 2018, pp. 520–535.

[170] C. Zhao, M. Shen, L. Sun, G.-Z. Yang, Generative localization with uncertainty estimation through video-ct data for bronchoscopic biopsy, IEEE Robotics and Automation Letters 5 (1) (2019) 258–265.

[171] L. Dong, C. Quirk, M. Lapata, Confidence modeling for neural semantic parsing, in: Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2018, pp. 743–753.

[172] A. Shen, D. Beck, B. Salehi, J. Qi, T. Baldwin, Modelling uncertainty in collaborative document quality assessment, in: Proceedings of the Workshop on Noisy User-generated Text, 2019, pp. 191–201.

[173] F. Pourpanah, D. Wang, R. Wang, C. P. Lim, A semisupervised learning model based on fuzzy min–max neural networks for data classification, Applied Soft Computing 112 (2021) 107856.

[174] M. Wu, L. Zhang, J. Niu, Q. M. J. Wu, Target detection in clutter/interference regions based on deep feature fusion for hfswr, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 14 (2021) 5581–5595.

[175] F. Pourpanah, R. Wang, C. P. Lim, X. Wang, M. Seera, C. J. Tan, An improved fuzzy artmap and q-learning agent model for pattern classification, Neurocomputing 359 (2019) 139–152.

[176] R. Wang, X.-Z. Wang, S. Kwong, C. Xu, Incorporating diversity and informativeness in multiple-instance active learning, IEEE Transactions on Fuzzy Systems 25 (6) (2017) 1460–1475.

[177] S. C. K. Shiu, D. S. Yeung, C. H. Sun, X. Z. Wang, Transferring case knowledge to adaptation knowledge: An approach for case-base maintenance, Computational Intelligence 17 (2) (2001) 295–314.

[178] J. N. K. Liu, Y.-L. He, E. H. Y. Lim, X.-Z. Wang, A new method for knowledge and information management domain ontology graph model, IEEE Transactions on Systems, Man, and Cybernetics: Systems 43 (1) (2013) 115–127.

[179] N. Zeng, D. Song, H. Li, Y. You, Y. Liu, F. E. Alsaadi, A competitive mechanism integrated multi-objective whale optimization algorithm with differential evolution, Neurocomputing 432 (2021) 170–182.

[180] W. Liu, Z. Wang, N. Zeng, Y. Yuan, F. E. Alsaadi, X. Liu, A novel randomised particle swarm optimizer, International Journal of Machine Learning and Cybernetics 12 (2) (2021) 529–540.

[181] W. Liu, Z. Wang, Y. Yuan, N. Zeng, K. Hone, X. Liu, A novel sigmoid-function-based adaptive weighted particle swarm optimizer, IEEE transactions on cybernetics.