Characterizing Reliability of Three-version Traffic Sign Classifier System through Diversity Metrics

Qiang Wen
Department of Computer Science
University of Tsukuba
Tsukuba, Japan
wen.qiang@sd.cs.tsukuba.ac.jp

Fumio Machida
Department of Computer Science
University of Tsukuba
Tsukuba, Japan
machida@cs.tsukuba.ac.jp

***Abstract*—The N-version machine learning (ML) system is an architecture approach to enhance the reliability of ML system outputs by exploiting ML model diversity and input data diversity. While existing studies theoretically show the relation between diversity metrics and system reliability, there is a shortage of empirical studies validating reliability models with diversity parameters in real datasets. In this paper, focusing on traffic sign recognition tasks, we empirically analyze the impact of diversity parameter estimations for predicting the reliability of three-version traffic sign classifier systems. Using five real-world traffic sign datasets, we confirm that the three-version architecture effectively enhances system reliability by applying diverse models and diversified input images. Then, we estimate the diversity parameters and apply them to variants of reliability prediction models. The prediction residuals between the observed reliability and the predicted reliability are mostly less than 0.017 across all data sets, which is half of the residual achieved by the conventional prediction model, except for the architecture of a single model with triple input. As the estimated values of diversity parameters tend to be stable with a relatively small number of samples, we consider that the reliability prediction models using diversity parameters are useful in the early-stage design of ML systems.**

Keywords—Classification, Diversity, Machine learning system, N-version programming, Reliability

I. INTRODUCTION

In recent years, machine learning (ML) techniques are developed rapidly and applied widely in various technical fields. Our daily lives tend to depend on ML-based intelligent software systems, such as face recognition, medical diagnosis, autonomous robots, and vehicles. As ML models used in such systems are trained from previous samples and are sensitive to new input data, inference errors on real input data are inevitable in many practical situations. However, incorrect outputs from ML models may cause serious problems if they are used in safety-critical systems. For instance, the misrecognition of traffic signs by ML model-based classifiers could lead to traffic accidents in autonomous driving [38]. Therefore, it is a stringent issue to secure the reliability of outputs from ML-based systems.

To provide reliable ML systems, multiple versions of ML models and input data sources are configured as a redundant system architecture, which is called N-version machine learning (ML) systems [10]. The reliability improvement by different architectures of N-version ML systems can be characterized by diversity metrics. Recent studies exploit diversity metrics to formulate the reliability of N-version ML system outputs [3][10]. Meanwhile, the diversity of ML models in terms of prediction errors in N-version ML systems is observed by measuring the mutual error rate [4] and the coverage of errors [19][20]. A recent study used the Gini coefficient and the Shannon equitability index to measure the diversity of ML models [21]. However, none of the existing works empirically disentangle the impacts of diversity due to ML model's ability and the diversity in different input data sources. It is known that the reliability of the architecture, like the double-model double-input (DMDI) system, is affected by both model diversity and input diversity [10]. Analyzing the joint impact of two types of diversities on the reliability of N-version ML systems with real data sets is a key challenge addressed in this work.

In this paper, to investigate the reliability improvement of N-version ML systems and the impact of diversity metrics on N-version ML system reliability, we conduct experiments with traffic sign recognition tasks using five real-world traffic sign datasets from different countries [22-26]. We configure three-version traffic sign classifier systems that consist of three ML modules for traffic sign classification and a majority voter for determining the final classification result. Each ML module may deploy the same ML model or diverse ML models. The three-version traffic sign classifier systems may also use diverse inputs. To create slightly different versions of input data, we apply image transformation methods, such as noise addition and image rotation. Depending on the choice of models and input data, the classifier system architectures can be divided into three types, namely Triple-Model Single-Input (TMSI), Single-Model Triple-Input (SMTI), and Triple-Model Triple-Input (TMTI). To predict the reliability improvement by the three-version architectures, we measure the model diversity and the input data diversity by the intersection of errors and the conjunction of errors, respectively, observed by the ML modules with test data sets. The estimated values are applied to the reliability models for three-version systems. Based on the existing models [3], we propose five variants for each reliability prediction model by considering different combinations of diversity parameters for a triple error probability. We validate the reliability prediction models by comparing the predicted reliability with the observed reliability for five data sets. The results show that the prediction residuals are less than 0.017, 0.07, and 0.012, for TMSI, SMTI, and TMTI architectures, respectively, across all data sets. We find that the variants of reliability prediction models are equally effective and considerably better than the reliability prediction by the conventional model [9] which assumes homogeneous dependence parameters among different ML

models. In summary, we give the following contributions in this paper.

1. Through the experiments with five real-world traffic sign datasets, we show that the reliability of the traffic sign classification system is improved by employing the three-version ML architecture.
2. We consider several variants of reliability models for three-version ML architectures and empirically show that they are equally effective to predict the reliability of three-version traffic sign classifier systems compared with the conventional N-version system reliability model.
3. We provide some findings from the empirical studies of diversity parameter estimations for reliability prediction, e.g., the number of samples required for estimating fairly stable diversity parameter values.

The remainder of the paper is organized as follows. Section II describes related work. Section III explains the reliability models used in the following experiment. Section IV clarifies the research questions addressed in the empirical study. Section V describes the experiment configuration. Section VI shows the results of the reliability analysis and gives answers to the research questions. Finally, Section VII describes the conclusion.

II. Related Work

To enhance the reliability of ML systems, various approaches have been studied, such as ML testing, data validation, safety monitors, and redundant architectures. ML testing focuses on identifying and resolving discrepancies between existing ML models and required conditions [27]. For instance, DeepXplore is an automated white-box testing approach that can detect incorrect behaviors in autonomous driving systems [28]. To detect real-world error-inducing corner cases at runtime, Deep Validation leverages the data validation approach that is also based on white box models for deep neural networks [29]. Several safety monitors have been presented for detecting out-of-distribution data at runtime [30][34]. However, such monitors need to be trained together with the ML model in advance. In contrast, the redundant architecture approaches [4][10][33] can achieve improved reliability through a simple redundancy scheme with diversity and does not require separate procedures for training monitors or white box models.

Many recent studies have investigated multi-version ML approaches to improve ML system reliability. N-version programming for ML components is revealed to have a huge potential to improve the overall reliability of ML components [2]. NV-DNN is proposed to improve the fault-tolerant ability of deep learning systems. It consists of N independently developed models and decision-making procedures [4]. Besides, a voting-based ensemble approach using multiple diverse machine learners is also used for improving the accuracy of intrusion detection [32]. Other studies show that diversified input data can also be used to improve the reliability of N-version ML systems [3] [19][20]. A multimodal deep learning approach is proposed to improve the classification accuracy of remote-sensing imagery, which achieves better results than single-model or single-modality approaches [31]. However, none of the above studies shows the estimations of diversity parameters and their impact on the N-version architecture reliabilities.

Several diversity measures have been presented for ensemble methods that are broadly divided into pairwise measures and non-pairwise measures [14]. To measure ensemble diversity, a classical approach is to measure the pairwise similarity/ dissimilarity between two learners and then average all the pairwise measurements for the overall diversity [14]. Disagreement measure [6], Q-statistic [15], correlation coefficient [16], and kappa-statistic [17] are different pairwise measures for diversity measurement. Among them, the correlation coefficient is a classic statistic for measuring the correlation between two binary vectors. Kappa-statistic is utilized to measure the diversity between two classifiers. For non-pairwise measures, interrater agreement [7], entropy [8], and coincident failure [18] are measures that estimate diversity directly. A recent study introduces the Gini coefficient and the Shannon equitability index to measure the diversity of ML models [21]. In this paper, the diversity estimation can be considered a pairwise measure. We focus on the joint impact of model diversity and input diversity that characterize the reliability of different architectures of N-version ML systems.

III. Reliability Model

This section introduces the reliability models and the diversity metrics that are examined in our empirical study.

A. Two-version and Three-version Machine Learning Systems

N-version ML architecture contains N different versions of ML modules that work for the same task at the same time. In N-version ML architectures, we can use multiple inputs and ML models in a system. As it is shown in Fig. 1, x_1, x_2 and x_3 represent the inputs to ML models from different data sources (e.g., sensors), while m_1, m_2 and m_3 represent different ML models dealing with the same task. Double model with double input system (DMDI) consists of two ML models m_1 and m_2 which receive inputs x_1 and x_2 respectively, voting on the inference results to determine the system output. Three-version architectures can be divided into three categories, which are TMSI, SMTI, and TMTI as shown in Fig. 1. In this paper, we focus on these four architectures.

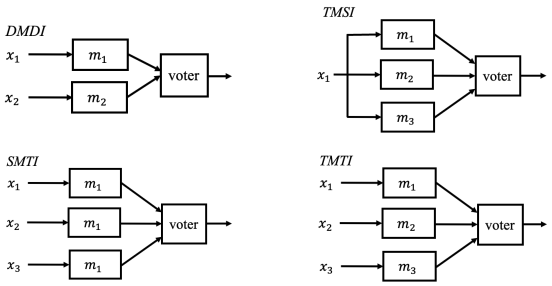

Fig. 1. Two-version and three-version architectures.

B. Conventional Reliability Model for N-version Systems

Unlike traditional hardware redundancy modules, the reliability of N-version systems cannot assume the complete independence of module failures [1]. Therefore,

a reliability model considering N-version programming is presented using a dependent failure parameter [9]. The parameter α measures the similarity percentage of the input sets on which each pair of versions fail. Assuming the failure probability of each version to be p, the reliability of a three-version system is given by

$$R = 1 - [3\alpha p(1-\alpha) + \alpha^2 p] = 1 - \alpha p(3 - 2\alpha). \quad (1)$$

Although this model can capture the dependence of modules, the ratio of the dependence is regarded as homogeneous which may not be true in reality. Moreover, the dependent failure parameter α is not enough to represent the dependence or similarity of input data for ML modules. To overcome the issues, we consider the reliability models using two diversity metrics.

C. Reliability Models using Model Diversity and Input Diversity

In this section, we review the reliability models in [3][19] and propose variants of the reliability models. We define system reliability as the probability that the output of the system is correct. We assume the correct output is determined by checking if the system's predicted label matches the correct label from the test datasets in the following sections. In real applications, the correct output refers to the output that aligns with the ground truth. Define $R_{\mathrm{i,j}}$ as the reliability of a ML system with i versions with j diversity inputs. For example, a system using a single model with single input can be given by $R_{1,1} = 1 - |E_k|/|S|$, where S represents the total sample space of inputs and $E_k \subseteq S$ is the set of input data that leads to output error by m_k.

Reliability models for N-version ML architectures introduce two types of diversity parameters that are model diversity and input diversity [19]. The model diversity is measured by the intersection of errors that is proportional to the ratio of input data sets that cause double errors of two ML models. The metric characterizes the diversity of ML models with respect to inference errors. The intersection of errors $\alpha_{i,j} \in [0,1]$ is formally defined as

$$\alpha_{i,j} = \frac{|E_i \cap E_j|}{\min\{|E_i|, |E_j|\}}, \quad (2)$$

where E_i represents the set of input data that leads to output error by ML model m_i. On the other hand, the input diversity is represented by the conjunction of errors that is the probability that an input data causes an error conditioned by the error with another input data. The metric characterizes the diversity of input data in terms of inference errors by the same ML model. The conjunction of errors $\beta_{i,s|t} \in [0,1]$ for ML model m_i is formally defined as the conditional probability

$$\beta_{i,s|t} = \Pr[x_s \in E_i | \, x_t \in E_i], \quad (3)$$

where x_s and x_t are input data from different data sources (e.g., different sensors). Using the two diversity metrics, the reliability of DMDI system can be formulated as follows.

$$R_{2,2}(m_1, m_2; x_1, x_2) = 1 - \left[\beta_{1,2|1} \cdot \alpha_{1,2} + (1 - \beta_{1,2|1}) \cdot \frac{p_2 - \alpha_{1,2} \cdot p_1}{(1 - p_1)}\right] \cdot p_1, \quad (4)$$

where p_i represents the error probability of ML models m_i with input x_1 (i.e., $\Pr[x_1 \in E_i]$).

While the existing study uses the diversity metrics to formulate the reliability of three-version ML architectures [3], we find that several alternative parameterizations for the reliability models are also possible. In the following, we propose five variants of reliability models for TMSI, SMTI, and TMTI systems.

1) *Reliability of TMSI*

Assuming that a TMSI system fails when more than two outputs are in error, the reliability of TMSI system using three ML models m_1, m_2 and m_3 is represented by

$$R_{3,1}(m_1, m_2, m_3; x_1) = 1 - (\alpha_{1,2}p_1 + \alpha_{1,3}p_1 + \alpha_{2,3}p_2 - 2\alpha_{1,2}\alpha_{1,3}p_1). \quad (5)$$

Variants of TMSI model are obtained with respect to the last term representing the probability of triple errors under the conditional independence assumption. The term $\alpha_{1,2}\alpha_{1,3}p_1$ can be replaced with other combinations, such as $\alpha_{1,2}\alpha_{2,3}p_1$ and $\alpha_{1,3}\alpha_{2,3}p_1$. Moreover, the arithmetic mean and the geometric mean of $\alpha_{1,2}\alpha_{1,3}p_1, \alpha_{1,2}\alpha_{2,3}p_1$, and $\alpha_{1,3}\alpha_{2,3}p_1$ can also give the estimates of triple error probability.

2) *Reliability of SMTI*

Assuming that an SMTI system also fails when more than two outputs are errors, the reliability of SMTI system using three inputs x_1, x_2 and x_3 is given by

$$R_{1,3}(m_1; x_1, x_2, x_3) = 1 - (\beta_{1,2|1}p_1 + \beta_{1,3|1}p_1 + \beta_{1,3|2}p_2{}' - 2\beta_{1,2|1}\beta_{1,3|1}p_1), \quad (6)$$

where $p_2{}'$ represents the error probability of ML models m_1 with input x_i (i.e., $\Pr[x_i \in E_1]$).

Similar to TMSI, we can consider variants of SMTI model with respect to the last term representing the probability of triple errors under the conditional independence assumption. In place of the term $\beta_{1,2|1}\beta_{1,3|1}p_1$, we can choose other combinations, such as $\beta_{1,2|1}\beta_{1,3|2}p_1$, $\beta_{1,3|1}\beta_{1,3|2}p_1$, the arithmetic and the geometric mean of $\beta_{1,2|1}\beta_{1,3|1}p_1$, $\beta_{1,2|1}\beta_{1,3|2}p_1$ and $\beta_{1,3|1}\beta_{1,3|2}p_1$ for representing the triple error probability.

3) *Reliability of TMTI*

Assuming that a TMTI system fails when at least two modules output errors, the reliability of TMTI system using three ML models m_1, m_2 and m_3 and three inputs x_1, x_2 and x_3 is given by

$$R_{3,3}(m_1, m_2, m_3; x_1, x_2, x_3) = 1 - [p_{2,2}(m_1, m_2; x_1, x_2) + p_{2,2}(m_1, m_3; x_1, x_3) + p_{2,2}(m_2, m_3; x_2, x_3) - 2p_{2,2}(m_1, m_2; x_1, x_2) \cdot p_{2,2}(m_1, m_3; x_1, x_3)/p_1], \quad (7)$$

where $p_{2,2}(m_{\mathrm{i}}, m_{\mathrm{j}}; x_{\mathrm{s}}, x_t)$ is the complement of DMDI reliability $1 - R_{2,2}(m_1, m_3; x_1, x_3)$ [3]. We can consider

variants of TMTI model with respect to the last term representing the probability of triple errors. The term $p_{2,2}(m_1, m_2; x_1, x_2) \cdot p_{2,2}(m_1, m_3; x_1, x_3)/p_1$ can be replaced with combinations, such as $p_{2,2}(m_1, m_2; x_1, x_2) \cdot p_{2,2}(m_2, m_3; x_2, x_3)/p_1$ and $p_{2,2}(m_1, m_3; x_1, x_3) \cdot p_{2,2}(m_2, m_3; x_2, x_3)/p_1$. Furthermore, the arithmetic mean and the geometric mean of these three terms can also give the estimates of triple error probability.

IV. Research Questions

As reviewed in the previous section, the reliability of N-version ML system architectures is theoretically investigated in relation to model diversity and input diversity. However, it has not been discussed how diversity metrics estimated from the empirical observations are effective for reliability prediction. In addition, it is also a question of which variant of the reliability model is the most suitable for predicting the reliability of three-version systems. Aiming to answer these questions, we conduct experiments on traffic sign recognition tasks by using deep neural networks for classifying various traffic signs from different countries. We empirically evaluate the reliability of three-version traffic sign classifier architectures and compare the results with the predicted reliability which is based on estimated diversity parameter values. In our empirical study, we consider the following research questions.

RQ1: Does the implementation of a three-version system architecture effectively enhance reliability?

RQ2: How can the reliability models using diversity parameters estimate well the reliability of traffic sign classifier architectures?

RQ3: How does the variant of the reliability models for three-version ML systems affect the reliability prediction performance?

RQ4: How many samples are required to obtain good estimates of the diversity parameter values?

RQ5: How does the different number of samples affect the sampling process and reliability predictions?

To address RQ1, we evaluate the reliability of three-version traffic sign classification systems using five different traffic sign datasets. We compare the reliability of the TMTI architecture with that of a single model employing a single input and observe the reliability improvement. To answer RQ2, we estimate the diversity parameters based on the outputs of three-version traffic sign classification systems. The estimated diversity parameters are used to derive the predicted reliability. We then compare the predicted reliability with the observed reliability to compute the prediction residuals. To answer RQ3, we use five variants of the reliability models for each three-version architecture and show their impacts on the prediction performance. For RQ4, we track the variances of diversity parameter values over the number of samples and analyze their trends. For RQ5, we conduct experiments with random sampling by varying the number of test samples and observe the changes in parameter estimations and reliability predictions.

V. Experiment Configuration

A. Classification Models

We adopt LeNet [11], AlexNet [12], and ResNet50 [13] for ML models for three-version traffic sign classification systems. They are well-known deep neural networks for image recognition tasks. Three networks are implemented on the TensorFlow platform and trained with traffic sign datasets (explained in the next subsection). We choose these models because their architectures are developed independently, and their performance is relatively balanced. Fig. 2 shows the inaccuracies (the ratio of classification errors) of eight neural networks trained with the Chinese Traffic Sign Dataset. We observe that the three models are equally competitive. Since the primary objective of this study is to investigate the impact of diversity parameter estimations on the system reliability instead of exploring the best combinations achieving the highest accuracy, we focus on the experiments with these three models.

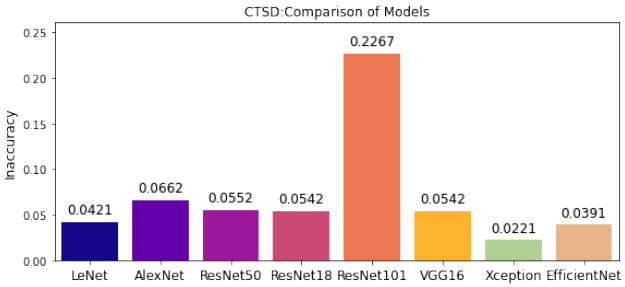

Fig. 2. Model inaccuracies for CTSD: LeNet, AlexNet, ResNet50, ResNet18, ResNet101, VGG16 [35], Xception [36], and EfficientNet [37].

B. Data Diversification

We use image transformations to generate the diversified input data. In real-world safety-critical scenarios such as autonomous driving, the sensors can receive traffic sign images with different resolutions, from different angles, or at different time points. We mimic such different versions of data by adding noise and rotating the original images. Fig. 3 shows the inaccuracies (the ratio of classification errors) of seven different inputs tested with LeNet in the Chinese Traffic Sign Dataset. The type of noise being added is Gaussian noise and the rotated images are generated by rotating the original images counterclockwise. We can see that when the variance of the noise is set to 0.01^2 and the rotation degree is 5, the inaccuracy closely resembles the original inaccuracy. Hence, we select noise 0.01^2 and rotate 5° input data. Fig. 4 shows the samples of diversified data, x_1 is the original, x_2 is noise-added and x_3 is rotated image data.

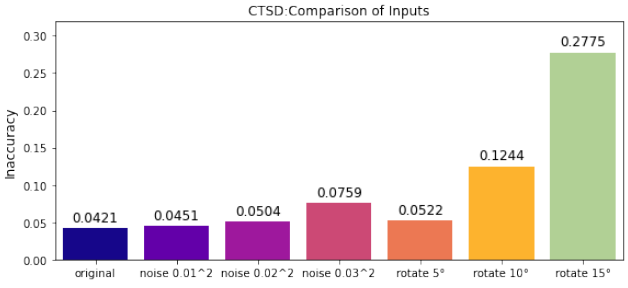

Fig. 3. Input inaccuracies for CTSD.

Fig. 4. Samples of three input data.

C. Datasets

We choose five different traffic sign datasets in the following experiment: the Chinese Traffic Sign Dataset (CTSD) [22], the German Traffic Sign Recognition Benchmark (GTSRB) [23], Traffic Sign Classification Dataset (TSCD) [24], Turkey Traffic Sign (TTS) [25] and Arabic Traffic Signs (ATS) [26]. LeNet, AlexNet, and ResNet50 are trained on CTSD, GTSRB, TSCD, TTS, and ATS with 4170, 34799, 58511, 11952, and 46200 training samples, respectively. The models are trained with 128 batch sizes and 20 epochs on all the datasets. Fig. 5 shows the convergence of the test accuracies of the three models for CTSD and GTSRB in 20 epochs.

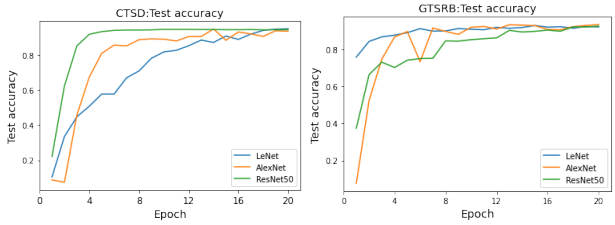

Fig. 5. The convergence trends of test accuracies for CTSD and GTSRB.

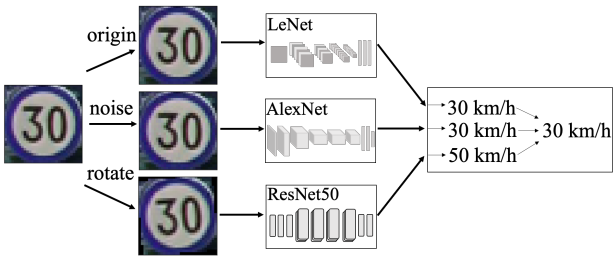

Fig. 6. A three-version system by TMTI architecture.

D. Three-version System Configuration

Fig. 6 shows an example configuration of a three-version system by TMTI architecture. We take the original, noise-added, rotated image data as inputs and feed them to three deep neural networks LeNet, AlexNet, and ResNet50. When recognizing the traffic sign like '30(km/h)', even though one of the ML models output errors like '50(km/h)', a voting decision from diversified prediction results can correct the error and avoid an undesirable result. The voting decision is based on the simple majority voting to be consistent with the model assumption in [3].

E. Evaluation Metric

The reliability of a three-version system is evaluated using test samples of traffic sign datasets. We measure the reliability by the ratio of correct outputs over the test samples, which we call the *observed reliability*. On the other hand, we can predict the reliability by the reliability models and the proposed variants in Section III using estimated values of diversity parameters. The validity of the reliability models can be evaluated by the *prediction residual e* which is the difference between the observed reliability R_{observed} and the predicted reliability $R_{\text{predicted}}$.

$$e = R_{observed} - R_{predicted}. \tag{8}$$

VI. Results

In this section, we denote the trained LeNet, AlexNet, and ResNet50 as $m_{\text{L}}, m_{\text{A}}$, and m_{R}, respectively; original data, noised data, and rotated data as input $x_{\text{o}}, x_{\text{n}}$, and x_{r}, respectively. First, we evaluate the accuracy of a single model with single input systems. The accuracy is calculated by the ratio of the number of misclassified samples to the number of total test samples. The numbers of test samples used for evaluations are 1994, 12630, 14628, 5313, and 9240, for CTSD, GTSRB, TSCD, TTS, and ATS, respectively. All the results are shown in TABLE I. For example, $R_{1,1}(m_L, x_o)$ represents the accuracy of LeNet with original data. We label the three highest accuracies in each dataset as bold. We can observe that the overall accuracy for TSCD is the highest and the accuracy for GTSRB is the lowest in general. While the accuracies are

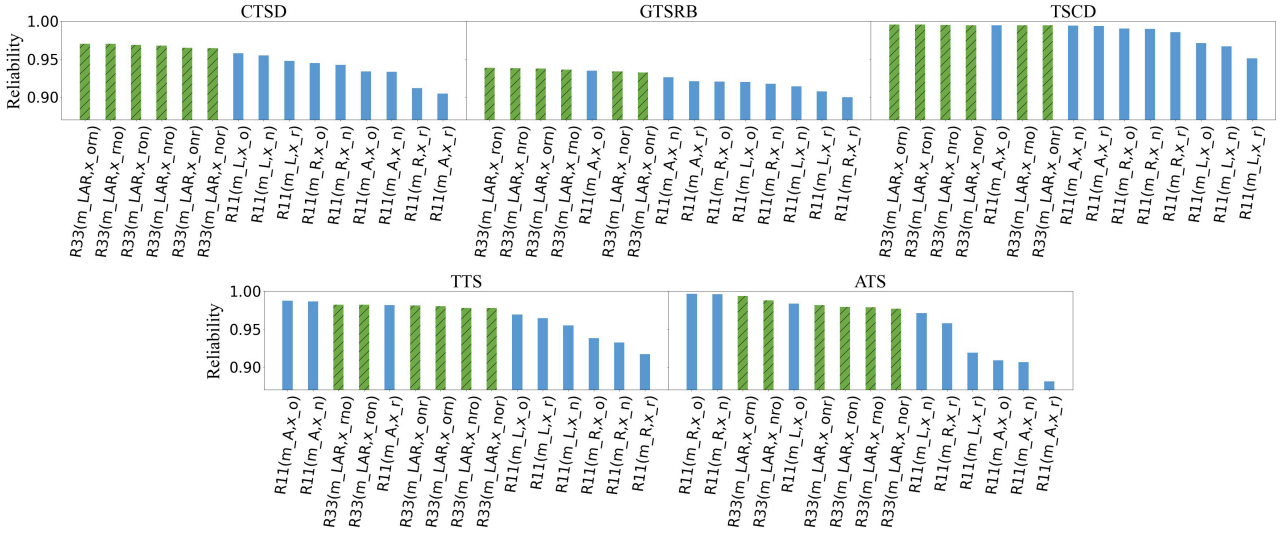

Fig. 7. TMTI reliability comparison.

generally balanced among different single-model systems, the accuracies for the ATS data set are relatively unbalanced which may impact the reliability of three-version architectures.

TABLE I. RELIABILITY OF SINGLE MODEL WITH SINGLE INPUT SYSTEM

	CTSD	GTSRB	TSCD	TTS	ATS
$R_{1,1}(m_L, x_o)$	**0.9579**	0.9199	0.9709	0.9691	**0.9837**
$R_{1,1}(m_A, x_o)$	0.9338	**0.9349**	**0.9947**	**0.9872**	0.909
$R_{1,1}(m_R, x_o)$	0.9448	0.9204	0.9900	0.9381	**0.9962**
$R_{1,1}(m_L, x_n)$	**0.9549**	0.914	0.9667	0.955	0.9712
$R_{1,1}(m_A, x_n)$	0.9333	**0.9263**	**0.9939**	**0.9864**	0.9064
$R_{1,1}(m_R, x_n)$	0.9423	0.9176	0.9895	0.9324	**0.9960**
$R_{1,1}(m_L, x_r)$	**0.9478**	0.9074	0.9508	0.9644	0.9188
$R_{1,1}(m_A, x_r)$	0.9047	**0.9207**	**0.9934**	**0.9814**	0.8811
$R_{1,1}(m_R, x_r)$	0.9117	0.8998	0.9852	0.9168	0.9577

We measure the reliability of single-model with single-input systems by the ratio of correct outputs over the test samples, which corresponds to the accuracy of individual classifiers shown in Table I. We use the reliability in Table I as the baseline to compare with the reliability of TMTI architecture. For TMTI architecture, there are 6 different configurations where we select 3 different inputs for 3 different ML models. The results are shown in Fig. 7. The six green bars (also labeled as *R33()*) in each dataset present the reliability of TMTI architectures. For example, $R33(m_{LAR}, x_{orn})$ represents the reliability of three-version system where LeNet with original input, AlexNet with rotated input and ResNet50 with noised input. The result indicates that the reliability of TMTI architectures outperforms all other single models in the CTSD dataset, but certain single models can achieve higher reliability in GTSRB, TSCD, TTS and ATS datasets. To provide a comprehensive ranking of the architectures, we calculate the sum of the ranks in every dataset as a score. The result reveals that the scores of six TMTI architectures are 15, 17, 21, 23, 31, 34, following (m_A, x_o) with a score of 36 and (m_R, x_o) with a score of 44, indicating that TMTI is the superior option over any single-model systems.

Observation 1. Three-version ML system architectures, especially the TMTI architecture, have the potential to efficiently improve system reliability compared to single models.

While TMTI is a favorable option for reliability improvement, the reliability enhancement clearly depends on the architecture choice. Therefore, the prediction of system reliability is necessary for selecting an effective architecture in practice. In the following sections, we will analyze the empirical estimates of diversity parameters for predicting the reliability of two-version and three-version ML systems.

A. Two-version Architecture

First, we evaluate the reliability of DMDI systems which can also be a component of TMTI systems. As presented in Section III, the reliability of DMDI systems is associated with model diversity and input diversity. Therefore, diversity parameter estimations impact the reliability prediction of DMDI systems as well. There are three different combinations of ML models by choosing two from LeNet, AlexNet, and ResNet50, and three different input data choices by choosing two from original, noised, rotated input data, resulting in 18 configurations in total, which are shown in TABLE II.

TABLE II. DIFFERENT CONFIGURATIONS OF DMDI SYSTEMS

1	$(m_L, m_A; x_o, x_n)$	7	$(m_L, m_R; x_o, x_n)$	13	$(m_A, m_R; x_o, x_n)$
2	$(m_L, m_A; x_o, x_r)$	8	$(m_L, m_R; x_o, x_r)$	14	$(m_A, m_R; x_o, x_r)$
3	$(m_L, m_A; x_n, x_o)$	9	$(m_L, m_R; x_n, x_o)$	15	$(m_A, m_R; x_n, x_o)$
4	$(m_L, m_A; x_n, x_r)$	10	$(m_L, m_R; x_n, x_r)$	16	$(m_A, m_R; x_n, x_r)$
5	$(m_L, m_A; x_r, x_o)$	11	$(m_L, m_R; x_r, x_o)$	17	$(m_A, m_R; x_r, x_o)$
6	$(m_L, m_A; x_r, x_n)$	12	$(m_L, m_R; x_r, x_n)$	18	$(m_A, m_R; x_r, x_n)$

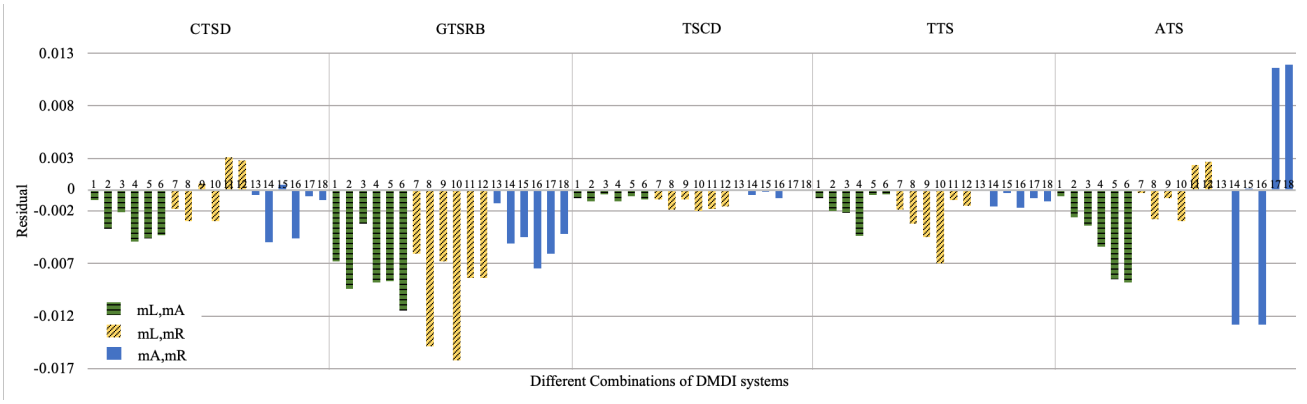

Fig. 8. DMDI residual between observed results and model results.

TABLE III. PARAMETERS FOR DIFFERENT DATASETS

CTSD		GTSRB		TSCD		TTS		ATS	
$\alpha_{L,A}$	0.4286	$\alpha_{L,A}$	0.4927	$\alpha_{L,A}$	0.3117	$\alpha_{L,A}$	0.4853	$\alpha_{L,A}$	0.2318
$\alpha_{L,R}$	0.1905	$\alpha_{L,R}$	0.5114	$\alpha_{L,R}$	0.2857	$\alpha_{L,R}$	0.3958	$\alpha_{L,R}$	0.3429
$\alpha_{A,R}$	0.2	$\alpha_{A,R}$	0.5328	$\alpha_{A,R}$	0.1299	$\alpha_{A,R}$	0.5	$\alpha_{A,R}$	0.4857
$\beta_{L,n\|o}$	0.9405	$\beta_{A,n\|o}$	0.9708	$\beta_{A,n\|o}$	0.7922	$\beta_{A,n\|o}$	0.9118	$\beta_{R,n\|o}$	0.9714
$\beta_{L,r\|o}$	0.6731	$\beta_{A,r\|o}$	0.7313	$\beta_{A,r\|o}$	0.5729	$\beta_{A,r\|o}$	0.6078	$\beta_{R,r\|o}$	0.0665
$\beta_{L,r\|n}$	0.7778	$\beta_{A,r\|n}$	0.8323	$\beta_{A,r\|n}$	0.75	$\beta_{A,r\|n}$	0.8824	$\beta_{R,r\|n}$	0.3333
$\beta_{R,r\|n}$	0.7778	$\beta_{R,r\|n}$	0.8769	$\beta_{R,r\|n}$	0.8333	$\beta_{L,r\|n}$	0.8824	$\beta_{L,r\|n}$	0.3333

We estimate the diversity metrics from all the test samples. For two-version ML systems, there are two diversity metrics to measure, i.e., $\alpha_{1,2}$ and $\beta_{1,2|1}$. The value of $\alpha_{1,2}$ is computed by the ratio of the number of test samples that cause double errors of two ML models m_1 and m_2. On the other hand, the value of $\beta_{1,2|1}$ is computed by the probability that input data causes an error conditioned by the error with the other input data. We apply the estimated values to expression (4) to predict reliability. The predicted reliability is then compared with the observed reliability. Fig. 8 shows the prediction residual for every configuration. We use green, yellow, and blue to represent combinations $(m_{\mathrm{L}}, m_{\mathrm{A}})$, $(m_{\mathrm{L}}, m_{\mathrm{R}})$, and $(m_{\mathrm{A}}, m_{\mathrm{R}})$, respectively. As we can see from Fig. 8, the residual of TSCD is generally smaller than the others, while that of GTSRB is larger than the others. The result implies that the residual of DMDI reliability prediction is related to the accuracies of individual ML models as observed in TABLE I. For CTSD, GTSRB, TSCD and TTS datasets where the accuracies are generally balanced among different single-model systems, higher overall model accuracies lead to lower residuals of DMDI reliability predictions. For the ATS data set, the residuals of the combination $m_{\mathrm{L}}, m_{\mathrm{R}}$ (yellow ones) are generally better than $m_{\mathrm{L}}, m_{\mathrm{A}}$ and $m_{\mathrm{A}}, m_{\mathrm{R}}$ (green and blue ones) since the accuracy of ML models $m_{\mathrm{L}}, m_{\mathrm{R}}$ is much higher than that of m_{A}. Overall, we observe that the absolute values of the residuals of DMDI reliability predictions are less than 0.017 across all datasets.

B. Three-version Architecture

Next, we evaluate the reliability of three-version architectures. We calculate the estimated diversity parameter values, which are shown in TABLE III. There are three different α values and eighteen different β values depending on the combinations of models and inputs. However, in TABLE III, we only show the β values used for reliability prediction. We compute the predicted reliability of TMSI, SMTI, and TMTI systems by applying the estimated diversity parameter values to the three-version reliability models. We also consider the baseline reliability model for three-version systems [9], which corresponds to expression (1). As the baseline model assumes that three models have the same reliability, we

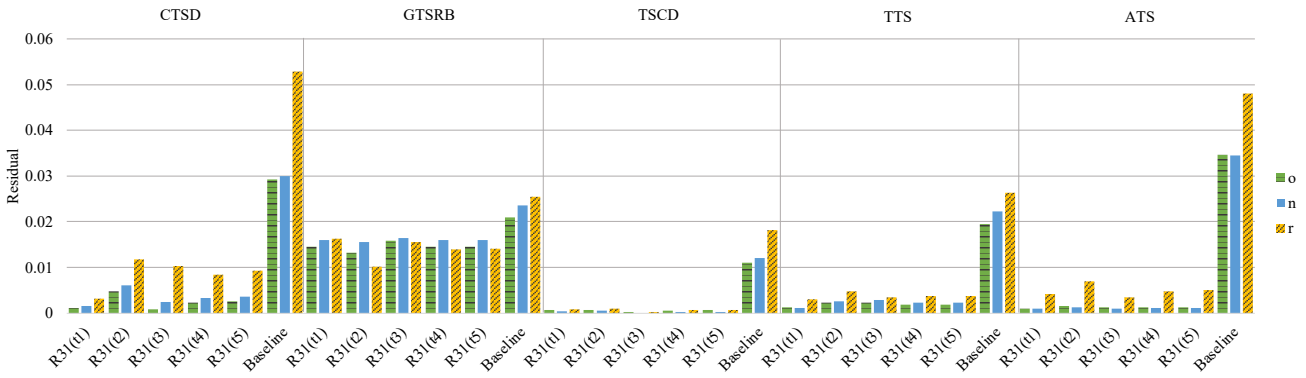

Fig. 9. Residual between observed results and model results for TMSI.

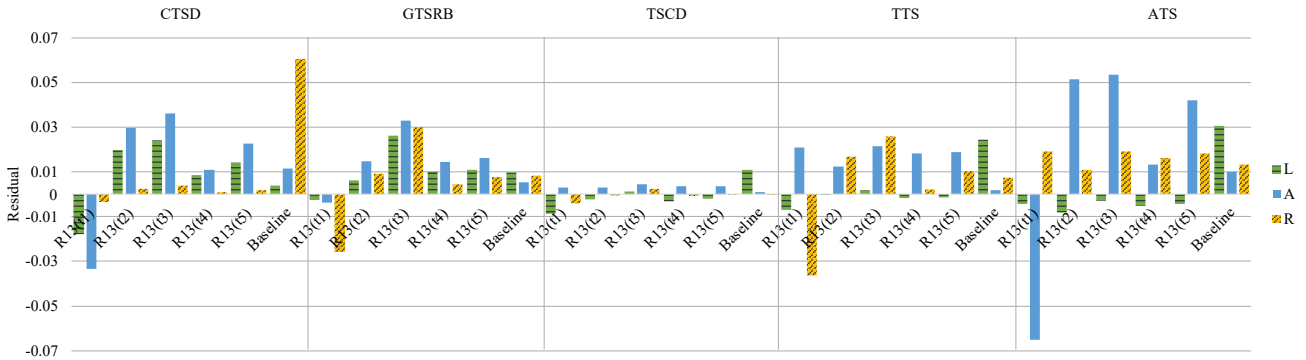

Fig. 10. Residual between observed results and model results for SMTI.

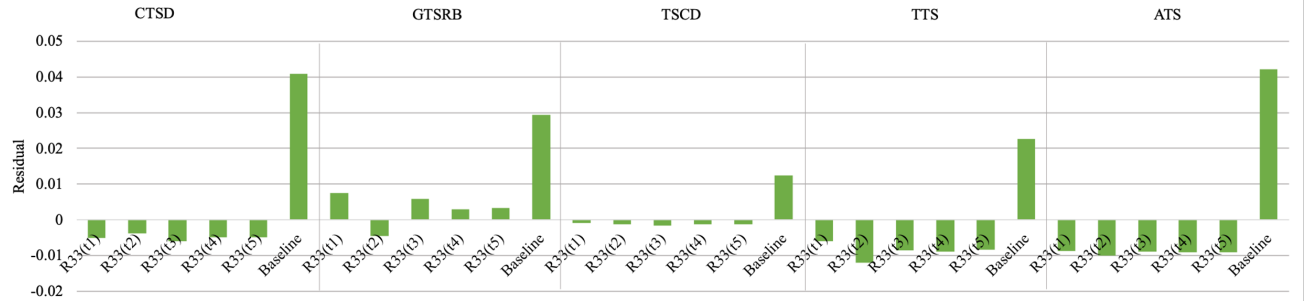

Fig. 11. Residual between observed results and model results for TMTI.

estimate the error probability p by the average error probabilities of three components. Similarly, the similarity percentage α is estimated by the average of three intersections of errors.

1) *TMSI*

TMSI architecture uses three ML models and single input. We apply ML models $m_{\mathrm{L}}, m_{\mathrm{A}}, m_{\mathrm{R}}$ and choose one input from $x_{\mathrm{o}}, x_{\mathrm{n}}, x_{\mathrm{r}}$. Hence there are three different situations $(m_L, m_A, m_R; x_{\mathrm{o}})$, $(m_L, m_A, m_R; x_{\mathrm{n}})$ and $(m_L, m_A, m_R; x_{\mathrm{r}})$. As mentioned in Section III.C, we can compute five variants of TMSI reliability prediction by different representations of triple error probability t_i:

$$\begin{aligned} t_1 &= \alpha_{1,2} \cdot \alpha_{1,3} \cdot p_1, \\ t_2 &= \alpha_{1,2} \cdot \alpha_{2,3} \cdot p_1, \\ t_3 &= \alpha_{1,3} \cdot \alpha_{2,3} \cdot p_1, \\ t_4 &= \frac{t_1 + t_2 + t_3}{3}, \\ t_5 &= \sqrt[3]{t_1 t_2 t_3}. \end{aligned} \tag{9}$$

We compute the residuals based on five predictions and the baseline, as shown in Fig. 9. For CTSD, GTSRB, TSCD and TTS datasets where the accuracies are generally balanced among different single-model systems, the residual of five variant predictions in TSCD is the smallest, then TTS, CTSD, and GTSRB is the largest.

TABLE I shows that the accuracy of TSCD is generally higher than the others, while GTSRB is the lowest among others. It illustrates that higher overall model accuracies lead to lower residuals of TMSI reliability predictions in datasets with balanced accuracy, which is consistent with our findings in DMDI. Besides, we find that the residuals of five predictions for each dataset are similar, and t_1 tends to provide better predictions. For TMSI architecture, the prediction residuals are less than 0.017 across all datasets. In addition, we find that the residual of a baseline is very large compared with the reliability predictions by three-version reliability models using diversity parameters. The reliability model for TMSI reduces the prediction residuals by 24.77%-100% from the prediction by the baseline model.

2) *SMTI*

SMTI architecture uses three inputs and a single ML model. We use inputs $x_{\mathrm{o}}, x_{\mathrm{n}}, x_{\mathrm{r}}$ and choose one ML model from m_L, m_A, m_R. Hence there are three different situations $(m_L; x_{\mathrm{o}}, x_{\mathrm{n}}, x_{\mathrm{r}})$, $(m_A; x_{\mathrm{o}}, x_{\mathrm{n}}, x_{\mathrm{r}})$ and $(m_R; x_{\mathrm{o}}, x_{\mathrm{n}}, x_{\mathrm{r}})$. Five SMTI reliability predictions mentioned in Section III.C are calculated by different representations of triple error probability t_i:

$$\begin{aligned} t_1 &= \beta_{1,2|1} \cdot \beta_{1,3|1} \cdot p_1, \\ t_2 &= \beta_{1,2|1} \cdot \beta_{1,3|2} \cdot p_1, \\ t_3 &= \beta_{1,3|1} \cdot \beta_{1,3|2} \cdot p_1, \\ t_4 &= \frac{t_1 + t_2 + t_3}{3}, \\ t_5 &= \sqrt[3]{t_1 t_2 t_3}. \end{aligned} \tag{10}$$

Then we calculate the residuals based on the above variants of SMTI models and the baseline. The result is shown in Fig. 10. The prediction residuals by five variants of the SMTI models in TSCD are still the smallest among other data sets. Compared with TSCD, other datasets have high residuals. Besides, we find that the residuals of five predictions for each dataset are similar, and t_4 tends to provide a better prediction. For SMTI architecture, the absolute values of the prediction residuals are less than 0.07 across five datasets. In addition, we find the residual of the baseline is generally similar to five predictions.

3) *TMTI*

TMTI architecture combines three inputs and three ML models. We use inputs $x_{\mathrm{o}}, x_{\mathrm{n}}, x_{\mathrm{r}}$ and models m_L, m_A, m_R. Similar to SMTI, we can compute five variants of TMTI reliability prediction mentioned in Section III.C by different representations of triple error probability t_i:

$$\begin{aligned} t_1 &= \frac{p_{2,2}(m_1, m_2; x_1, x_2) \cdot p_{2,2}(m_1, m_3; x_1, x_3)}{p_1}, \\ t_2 &= \frac{p_{2,2}(m_1, m_2; x_1, x_2) \cdot p_{2,2}(m_2, m_3; x_2, x_3)}{p_1}, \\ t_3 &= \frac{p_{2,2}(m_1, m_3; x_1, x_3) \cdot p_{2,2}(m_2, m_3; x_2, x_3)}{p_1}, \\ t_4 &= \frac{t_1 + t_2 + t_3}{3}, \\ t_5 &= \sqrt[3]{t_1 t_2 t_3}. \end{aligned} \tag{11}$$

We evaluate the residuals based on the above variants of TMTI models and the baseline, which are shown in Fig. 11. Similar to TMSI and SMTI, the prediction residuals by five variants of TMTI models in TSCD are still the smallest among other data sets. Besides, we observe that the residuals of five predictions for each dataset are almost the same and cannot figure out which variant is better. For TMTI architecture, the absolute values of the prediction residuals are less than 0.012 across five datasets. In addition, we find that the residual of the baseline is the largest compared with five variant predictions for each dataset. The reliability model for TMTI reduces the prediction residual by 47.3%-92.75% from the baseline prediction.

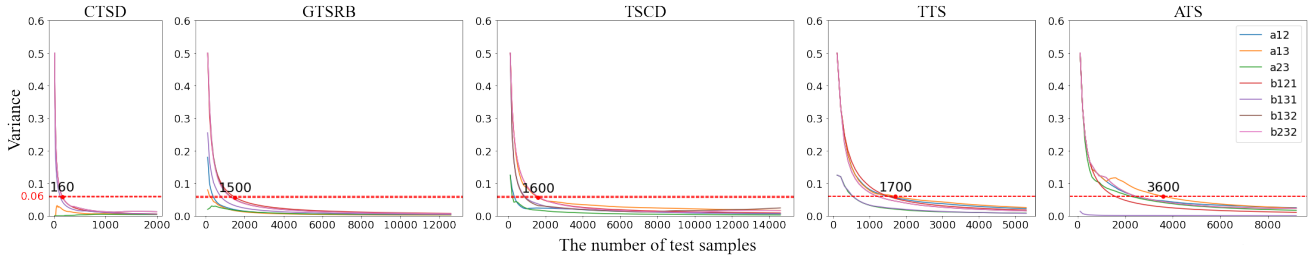

Fig. 12. The trends of variances of estimated diversity parameters over the number of samples.

In summary, the evaluation results for three different architectures give the following observations that correspond to the answers to RQ2 and RQ3.

Observation 2. The prediction residuals are mostly less than 0.017 across five data sets in most architectures except the SMTI architecture.

Observation 3. The residuals of five variants of TMSI, SMTI, and TMTI reliability predictions are equally effective. No variant shows evident superiority over the others.

C. Trends of Diversity Parameter Estimations

To answer RQ4, we further investigate how diversity parameters change over the number of samples observed. We compute the variances of the estimated diversity parameters over the course of sample observations. The trends of variances of the estimated diversity parameter values are plotted in Fig. 12. For all datasets, we can see that the variances tend to decrease drastically after a certain number of samples, indicating that the estimated values tend to converge after receiving a sufficient number of samples. Fig. 12 also indicates the number of samples necessary to reach a threshold of variance. For instance, if we set the threshold to 0.06, at least 160 samples are necessary to obtain all the diversity estimates satisfying the criterion in CTSD. For GTRSB, about 1500 samples are necessary to reach the threshold. The results imply that we can obtain fairly good estimates of diversity parameters with early observed samples for predicting and comparing the reliabilities of N-version system architectures.

Observation 4. For some data sets, we can obtain fairly good estimates of diversity parameters by a relatively small number of samples (less than a few thousand samples). In such cases, we may predict the reliability of three-version systems by measuring the diversities from early samples.

D. Impact of the Number of Samples

To further analyze the impact of sample size, we conduct experiments with random sampling from the test data set. We use GTSRB and TSCD for this experiment as they contain a relatively large number of samples. We fix the sample size n to 500, 750, and 1000, and randomly choose n samples from the test data set to predict the reliability. The average and the 95% confidence intervals of the predicted reliabilities of TMTI (R33(t1)) over ten trials are shown in TABLE IV. As the sample size increase, the prediction performance generally improves as its residual decreases and the confidence interval narrows.

TABLE IV. RELIABILITY PREDICTIONS BY RANDOM SAMPLING

Data set	Sample size	Average and CI	Residual
GTSRB	500	0.9288 [0.9276, 0.9300]	0.0098
	750	0.9291 [0.9283, 0.9289]	0.0082
	1000	0.9361 [0.9356, 0.9366]	0.0076
TSCD	500	0.9961 [0.9959, 0.9963]	0.0011
	750	0.9956 [0.9954, 0.9959]	0.0003
	1000	0.9954 [0.9952, 0.9955]	0.0002

Observation 5. The number of samples impacts the sampling process and the reliability prediction. A larger sample size generally provides a better prediction.

E. Discussion

Suggestions for reliable ML system design. The observations from our empirical study give some guidance for system designers and researchers who are considering N-version ML systems for designing reliable ML systems. While the estimation of reliability based on the accuracy over test samples may not perfectly reflect reliability in real-world use, it provides valuable insights into the system's performance and capabilities. From observation 1, we recommend implementing a three-version architecture since three-version architectures especially the TMTI architecture are efficient in improving the system reliability. From observations 2 and 3, we can suggest using the reliability models to choose the most reliable three-version architecture based on the observed diversities. It is advisable to choose SMTI in terms of cost. Although the residuals for SMTI are slightly worse than others, they are still better than the predictions by the conventional model. We can improve the reliability without training and deploying diverse models when deploying SMTI unless it has a significant disadvantage in the predicted reliability. From observations 4 and 5, collecting as many samples as possible is recommended to make accurate reliability predictions. However, for the architecture comparison purpose, a relatively small number of samples may be satisfactory for obtaining reasonable estimates of diversity parameters. In the early stage of system design or in system testing with real samples, it is worth evaluating the reliabilities with available samples to choose a suitable architecture.

Threats and limitations. In this study, we focus on traffic sign image recognition tasks to evaluate the reliability of three-version ML systems. The observations are mostly consistent across the five data sets we adopted. However, for comparison purposes, we fixed three ML models (LeNet, AlexNet, and ResNet50) and the diversification methods for input data (original, noise-added, and rotated). Different ML models and other data diversification methods may impact the results. Other empirical studies using various models [4][32] and data diversification methods [20] can complement our observations. Other tasks like object detection which share similarities with classification tasks could potentially benefit from adapting N-version ML system architectures. Some of our findings can be transferable to other ML tasks with neural networks. Nonetheless, we recognize that decision schemes and voting rules for tasks like regression may require further investigation. The presented study is limited to three-version systems using majority voting which is also the limitation of the theoretical analysis of N-version ML system reliability [3]. It is an important future work to evaluate the reliability of N-version systems with more versions and other voting schemes such as weighted voting both theoretically and experimentally. In this study, we did not consider other system design factors, such as performance, resource consumption, energy, and cost that also need to be considered together with reliability [5]. Since the associated costs and performance overhead depend on the chosen architecture, it is an important design challenge to find the best option under the given constraints that are considered in our future work.

VII. Conclusion

In this paper, we investigated the reliability of N-version ML systems and the associated diversity metrics estimated from the empirical data. We focused on traffic sign recognition tasks and conduct experiments on five different traffic sign datasets. We demonstrated the superiority of three-version ML system architectures, especially TMTI architecture, in terms of reliability improvement. However, it is important to note that TMTI does not consistently outperform other architectures across all five datasets. Therefore, we used reliability prediction models for three-version ML systems to compare the architecture reliabilities. The experiment results showed that the prediction residuals are mostly less than 0.017 in most architectures except SMTI architecture. While we considered five variants of reliability prediction models, it was shown that the five variants are almost equally effective. Moreover, through the trend analysis of estimated diversity parameter values, we observe that fairly good estimates can be obtained by a relatively small number of samples. The result implies that the diversity parameter estimations with early samples are useful for predicting the three-version ML system reliability and choosing the effective architecture. In future work, we will explore other ML tasks and the cost and performance of N-version ML systems.

Acknowledgment

This work was partly supported by JST SPRING, Grant Number JPMJSP2124, and JSPS KAKENHI Grant Number 22K17871. We would thank Alexandre Sparton for his help in our preliminary experiments.

References

[1] L. Chen and A. Avizienis, N-version programming: A fault-tolerance approach to reliability of software operation, In Proc. of 8th IEEE Int. Symp. on Fault-Tolerant Computing (FTCS-8), pp. 3-9, 1978.

[2] A. Gujarati, S. Gopalakrishnan and K. Pattabiraman, New wine in an old bottle: N-version programming for machine learning components, In Proc. of IEEE International Symposium on Software Reliability Engineering Workshops, pp. 283-286, 2020.

[3] Q. Wen, F. Machida, Reliability Models and Analysis for Triple-model with Triple-input Machine Learning Systems, In Proc. of the 5th IEEE Conference on Dependable and Secure Computing, pp. 1-8, 2022.

[4] H. Xu, Z. Chen, W. Wu, Z. Jin, S. Kuo, M. R. Lyu, NV-DNN: towards fault-tolerant DNN systems with N-version programming, In Proc. of the 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W), pp. 44-47, 2019.

[5] Y. Makino, T. Phung-Duc and F. Machida, A Queueing Analysis of Multi-model Multi-input Machine Learning Systems, In Proc. of the 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W), pp. 141–149, 2021.

[6] D. B. Skalak, The sources of increased accuracy for two proposed boosting algorithms, In Proc. American Association for Artificial Intelligence, AAAI-96, Integrating Multiple Learned Models Workshop, pp. 1133, 1996.

[7] J. L. Fleiss, B. Levin, and M. C. Paik, Statistical Methods for Rates and Proportions, 1st ed. Wiley, 2003. doi: 10.1002/0471445428.

[8] P. Cunningham and J. Carney, Diversity versus quality in classification ensembles based on feature selection, In European Conference on Machine Learning, pp. 109-116, 2000.

[9] M. Ege, A. Eyler M and MU. Karakas, Reliability analysis in N-version programming with dependent failures, In Proc. of IEEE EUROMICRO Conference, pp. 174-181, 2001.

[10] F. Machida, N-version machine learning models for safety critical systems, In Proc. of the DSN Workshop on Dependable and Secure ML, pp. 48-51, 2019.

[11] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In Proc of the IEEE, 1998.

[12] A. Krizhevesky, I. Sutskever, and G. E. Hinton, Imagenet classification with deep convolutional neural networks, Communications of the ACM, pp. 84-90, 2017.

[13] K. He, X. Zhang, S. Ren and J. Sun, Deep residual learning for image recognition, In Proc. of the IEEE conference on computer vision and pattern recognition, pp. 770-778, 2016.

[14] Z H. Zhou, Ensemble methods: foundations and algorithms[M]. CRC press, 2012.

[15] G U. Yule, VII. On the association of attributes in statistics: with illustrations from the material of the childhood society, Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character, Vol. 194(252-261), pp. 257-319, 1900.

[16] P H A. Sneath, The principles and practice of numerical classification[J]. Numerical taxonomy, 1973.

[17] D. D. Margineantu, T. G. Dietterich, Pruning adaptive boosting, In ICML, vol. 97, pp. 211-218, 1997.

[18] D Partridge, W. Krzanowski, Software diversity: practical statistics for its measurement and exploitation, Information and software technology, pp. 707-717, 1997.

[19] F. Machida, On the diversity of machine learning models for system reliability, IEEE Pacific Rim Int'l Symp. on Dependable Computing (PRDC), pp. 276-285, 2019.

[20] M. Takahashi, F. Machida, and Q. Wen, How Data Diversification Benefits the Reliability of Three-Version Image Classification Systems, IEEE Pacific Rim Int'l Symp. on Dependable Computing (PRDC), pp. 34-42, 2022.

[21] A. Chan, N. Narayanan, A. Gujarati, K. Pattabiraman, S. Gopalakrishnan, Understanding the Resilience of Neural Network Ensembles against Faulty Training Data, In Proc. of 21st International Conference on Software Quality, Reliability and Security (QRS), pp. 1100-1111, 2021.

[22] National Nature Science Foundation of China (NSFC). (2020). Database home. www.nlpr.ia.ac.cn. http://www.nlpr.ia.ac.cn/pal/trafficdata/index.html.

[23] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, The German Traffic Sign Recognition Benchmark: A multi-class classification competition, In Proc. of the IEEE International Joint Conference on Neural Networks, pp. 1453-1460, 2011.

[24] https://www.kaggle.com/datasets/flo2607/traffic-signs-classification

[25] https://www.kaggle.com/datasets/erdicem/traffic-sign-images-from-turkey

[26] G. Latif, J. Alghazo, D. A. Alghmgham. L. Alzubaidi, ArTS: Arabic Traffic Sign Dataset, Mendeley Data, V1, doi: 10.17632/4tznkn45mx.1, 2020.

[27] J. M. Zhang, M. Harman, L. Ma, and Y. Liu, Machine learning testing: Survey, landscapes and horizons, IEEE Transactions on Software Engineering, 2020.

[28] K. Pei, Y. Cao, J. Yang, and S. Jana, DeepXplore: Automated whitebox testing of deep learning systems, In Proc. of the 26th Symposium on Operating Systems Principles, pp. 1–18, 2017.

[29] W. Wu, H. Xu, S. Zhong, M. Lyu, and I. King, Deep validation: Toward detecting real-world corner cases for deep neural networks, In Proc. of the 49th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 125-137, 2019.

[30] R. S. Ferreira, J. Arlat, J. Guiochet, and H. Waselynck, Benchmarking safety monitors for image classifiers with machine learning, In Proc. of IEEE Pacific Rim International Symposium on Dependable Computing (PRDC), pp. 7-16, 2021.

[31] D. Hong, L. Gao, N. Yokoya, J. Yao, J. Chanussot, Q. Du, B. Zhang, More diverse means better: Multimodal deep learning meets remote-sensing imagery classification, IEEE Transactions on Geoscience and Remote Sensing, vol. 59(5), pp. 4340-4354, 2020.

[32] T. Zoppi, A. Ceccarelli, A. Bondavalli, Detecting Intrusions by Voting Diverse Machine Learners: Is It Really Worth?, IEEE Pacific Rim Int'l Symp. on Dependable Computing (PRDC), pp. 57-66, 2021.

[33] S. Latifi, B. Zamirai and S. Mahlke. PolygraphMR, Enhancing the reliability and dependability of CNNs, In Proc. of 50th IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), pp. 99-112, 2020

[34] J. Guerin, R. S. Ferreira, K. Delmas, J. Guiochet, Unifying evaluation of machine learning safety monitors, In Proc. of IEEE 33rd International Symposium on Software Reliability Engineering (ISSRE), pp. 414-422, 2022.

[35] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. Int. Conf. on Learning Representations, San Diego, CA, 2015.

[36] F. Chollet, Xception: Deep learning with depthwise separable convolutions, In Proc. of the IEEE conference on computer vision and pattern recognition, pp. 1251-1258, 2017.

[37] M. Tan, Q. Le, Efficientnet: Rethinking model scaling for convolutional neural networks. In International conference on machine learning, pp. 6105-6114, 2019.

[38] J. Wang, L. Shi, Y. Zhao, H. Zhang and E. Szczerbicki, Adversarial attack algorithm for traffic sign recognition, Multimedia Tools and Applications, pp. 1-13, 2022.