Uncertainty-aware Evaluation of Time-Series Classification for Online Handwriting Recognition with Domain Shift

Andreas Klaß[1,2,*], **Sven M. Lorenz**[1,2,*], **Martin W. Lauer-Schmaltz**[4], **David Rügamer**[2,3], **Bernd Bischl**[2], **Christopher Mutschler**[1] and **Felix Ott**[1,2]
[1]Fraunhofer IIS, Fraunhofer Institute for Integrated Circuits IIS
[2]LMU Munich, Munich, Germany
[3]RWTH Aachen, Aachen, Germany
[4]Technical University of Denmark
{a.klass, sven.lorenz}@campus.lmu.de, {david.ruegamer, bernd.bischl}@stat.uni-muenchen.de,
{christopher.mutschler, felix.ott}@iis.fraunhofer.de, {mwola}@dtu.dk

Abstract

For many applications, analyzing the uncertainty of a machine learning model is indispensable. While research of uncertainty quantification (UQ) techniques is very advanced for computer vision applications, UQ methods for spatio-temporal data are less studied. In this paper, we focus on models for online handwriting recognition, one particular type of spatio-temporal data. The data is observed from a sensor-enhanced pen with the goal to classify written characters. We conduct a broad evaluation of aleatoric (data) and epistemic (model) UQ based on two prominent techniques for Bayesian inference, Stochastic Weight Averaging-Gaussian (SWAG) and Deep Ensembles. Next to a better understanding of the model, UQ techniques can detect out-of-distribution data and domain shifts when combining right-handed and left-handed writers (an underrepresented group).

1 Introduction

Traditional machine learning (ML) algorithms assume training and test datasets to be *independently and identically distributed* [Sun *et al.*, 2016; Schölkopf *et al.*, 2021]. For many real-world applications, data often changes over time and space, and hence, training and test data originate from different distributions. This can cause ML models to fail due to a *domain shift* between training and test data [Sun *et al.*, 2016]. Transfer learning [Pan and Yang, 2009; Shao *et al.*, 2014] and domain adaptation [Long *et al.*, 2014; Saenko *et al.*, 2010] techniques can compensate for this domain shift. A first step in adapting for this domain shift is its detection, e.g., by having reliable uncertainty estimates of the model predictions [Li *et al.*, 2022]. Thus, to estimate the uncertainty of the model, a robust uncertainty quantification (UQ) technique is required that runs in real-time.

[*] Equal contribution

Approximate Bayesian Inference Techniques. In the field of deep learning (DL), UQ has lately seen a steep increase in interest. Recently, many promising methods have been proposed such as Variational Online Gauss-Newton (VOGN) [Khan *et al.*, 2018], Stochastic Weight Averaging-Gaussian (SWAG) [Maddox *et al.*, 2019], Bayes by Backpropagation (BBB) [Blundell *et al.*, 2015], and Laplace Approximation [Daxberger *et al.*, 2021]. Another widely used technique are Deep Ensembles [Lakshminarayanan *et al.*, 2017], which often yield well-calibrated models while being relatively easy to implement.

Decomposing Uncertainty. Several ways for estimating and decomposing uncertainty have been proposed. A common distinction is made between *aleatoric* uncertainty, which refers to the variability in the data, and *epistemic* uncertainty, which is the model's uncertainty caused by a lack of knowledge [Hüllermeier and Waegeman, 2021]. Building on [Kendall and Gal, 2017], [Kwon *et al.*, 2018] argue that neural networks (NNs) for classification are basically generalized linear models with error structure of multinomial and composite link functions. Hence, to acknowledge that the variance of a multinomial outcome is a function of the mean outcome, they propose to directly compute the variability in the softmax outputs. Another method to dissect total predictive uncertainty has been put forward by [Smith and Gal, 2018] and similarly by [Depeweg *et al.*, 2018] who propose to extract epistemic and aleatoric uncertainties from the predictive distribution of a Bayesian NN by calculating the entropy and mutual information. For an extensive survey of related approaches, see [Gawlikowski *et al.*, 2021].

UQ for OnHW. UQ techniques have been broadly evaluated in computer vision applications such as image classification [Kendall and Gal, 2017], i.e., optical character recognition (OCR), but methods have rarely been evaluated on spatio-temporal datasets [Cai *et al.*, 2014]. OCR is concerned with offline handwriting recognition from images. In contrast, online handwriting (OnHW) recognition works on different types of spatio-temporal signals and can make use of temporal information such as writing speed and direction [Plamondon and Srihari, 2000]. While many recording sys-

tems make use of a stylus pen together with a touch screen surface, sensor-enhanced pens, e.g., [Ott *et al.*, 2020; 2022a; 2022c; 2022d], based on inertial measurement units (IMUs) enable new applications. These pens stream data from accelerometer, gyroscope, magnetometer and force sensors in real-time represented as spatio-temporal multivariate time-series (MTS). The advantage of exploiting this temporal information is the ability to better distinguish between similarly shaped letters from dynamic information (number of strokes etc.). Spatio-temporal data can further help to identify certain characteristics in the data. [Ott *et al.*, 2022b], e.g., showed the domain shift between right-handed and left-handed writers by analyzing feature embeddings of their model for OnHW data.

Contribution. In this paper we evaluate the uncertainty of OnHW model predictions with SWAG [Maddox *et al.*, 2019] and Deep Ensembles [Lakshminarayanan *et al.*, 2017] for spatio-temporal reasoning, assessment of out-of-distribution detection, and pattern and failure recognition. We use uncertainty decompositions based on the method by [Kwon *et al.*, 2018] and [Smith and Gal, 2018] to evaluate the UQ techniques. Our claims are further supported by utilizing confidence and accuracy metrics to estimate the expected calibration error (ECE) [Guo *et al.*, 2017]. For an OnHW task with domain shift between right- and left-handed writers, we evaluate uppercase, lowercase and combined character classification tasks. Our source code will be available upon publication.[1]

The remainder of the paper is organized as follows. Section 2 discusses related work. In Section 3, we describe the background of Bayesian modeling and approximate inference. The experimental setup is described in Section 4, and results are discussed in Section 5.

2 Related Work

We first present related work of UQ with focus on spatio-temporal reasoning in Section 2.1. Section 2.2 summarizes state-of-the-art results for OnHW recognition.

2.1 UQ for Spatio-Temporal Reasoning

[Wu *et al.*, 2021] analyzed Bayesian and frequentist UQ methods for spatio-temporal forecasting on network traffic, epidemics and air quality datasets. Their evaluation shows that Bayesian methods are typically more robust in mean prediction, while confidence levels from frequentist methods provide better coverage over data variations (i.e., out-of-distribution data). Furthermore, traditional learning schemes lack knowledge about uncertainty. STUaNet [Zhou *et al.*, 2021] addresses this issue for spatio-temporal human mobility forecasting by injecting controllable uncertainty. This allows insights to both, UQ and weak supervised learning. [Gómez *et al.*, 2021] focused on the spatio-temporal uncertainty of urban prediction (where and when a piece of land becomes urban). [Li *et al.*, 2022] argue that the feature statistics such as mean and standard deviation (the domain characteristics of the training data), can be manipulated to improve the generalizability of DL models by modeling the uncertainty of domain shifts with feature statistics during training (that follow a multivariate Gaussian distribution). In the context of domain adaptation, [Cai *et al.*, 2014] adressed the extraction of domain-invariant representations for MTS classification.

2.2 Online Handwriting Recognition

[Ott *et al.*, 2020] initially proposed the *OnHW-chars* dataset and evaluated machine and DL techniques for the OnHW MTS classification task. The dataset contains right-handed and left-handed writers with a domain shift between both groups of writers (i.e., domains). [Ott *et al.*, 2022b] showed that transfer learning from small adaptation datasets results in poor model performances. Hence, their domain adaptation approach transforms features from left-handed writers into the domain of features from right-handed writers by optimal transport techniques. A reliable UQ method could identify out-of-distribution samples and only apply the transformation on samples for which the model has a high uncertainty. [Ott *et al.*, 2022a] combined offline and online handwriting recognition with a cross-modal representation learning technique by increasing the dataset size by using generative models. A robust uncertainty estimation technique could select samples with high model uncertainty.

3 Methodological Background

In the following we describe Bayesian model averaging in Section 3.1 and the two employed Bayesian UQ methods in Section 3.2. The decomposition of total predictive uncertainty into aleatoric and epistemic uncertainty is discussed in Section 3.3.

3.1 Bayesian Model Averaging

Bayesian approaches in DL naturally represent uncertainty by placing a distribution over model parameters and then marginalizing these parameters to form a predictive distribution (*Bayesian model averaging*) [Hoeting *et al.*, 1999]. Let $p(\theta|D)$ be the posterior distribution over model parameters θ, i.e., real-valued weights in the NN, given training dataset D, and let $p(y^*|x^*, \theta)$ denote the probability distribution over model outputs y^* (predicted classes), given sample x^*, and model weights θ. For the OnHW classification task, the sample x^* is an MTS $\mathbf{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_q\} \in \mathbb{R}^{q \times l}$, an ordered sequence of $l = 13$ streams with $\mathbf{u}_i = (u_{i,1}, \ldots, u_{i,l}), i \in \{1, \ldots, q\}$, where $q = 64$ is the length of the MTS. The training set D is a subset of the array $\mathcal{U} = \{\mathbf{U}_1, \ldots, \mathbf{U}_{n_U}\} \in \mathbb{R}^{n_U \times q \times l}$, where n_U is the number of time-series. The aim is to predict an unknown class label $y^* \in \mathcal{Y}$ with K classes (i.e., character labels) for a given MTS. The predictive distribution of the target variable is then given by

$$p(y^*|x^*, D) = \int p(y^*|x^*, \theta) p(\theta|D) d\theta. \quad (1)$$

In practice, we can approximate this integral by drawing S Monte Carlo samples from the posterior distribution:

$$p(y^*|x^*, D) \approx \frac{1}{S} \sum_{s=1}^{S} p(y^*|x^*, \theta_s) \quad , \quad \theta_s \sim p(\theta|D). \quad (2)$$

[1] Code and datasets: www.iis.fraunhofer.de/de/ff/lv/dataanalytics/anwproj/schreibtrainer/onhw-dataset.html

The predicted probability of an outcome is thus a weighted average over its probabilities with the weights being determined by $p(\theta|D)$.

3.2 Approximate Bayesian Inference

In order to apply Bayesian inference to an NN, we need to compute the posterior $p(\theta|D)$ of the NN weights. As the computation of the posterior is usually intractable, a (local) approximation is often used. This can be addressed by SWAG and Deep Ensembles with the latter abstaining from explicitly modeling $p(\theta|D)$ – nevertheless, this method can be considered to be in the field of approximate Bayesian inference.

Stochastic Weight Averaging-Gaussian (SWAG). SWAG [Maddox *et al.*, 2019] is a Bayesian inference technique for DL that builds on Stochastic Weight Averaging (SWA) [Izmailov *et al.*, 2018]. SWA computes an average of stochastic gradient decent (SGD) iterates to obtain information about the geometry of $p(\theta|D)$ from its trajectory. This posterior is then approximated by a Gaussian with simplified covariance structure and reduced dimensionality.

Deep Ensembles. Deep Ensembles are a committee of individual NNs initialized with a different seed [Lakshminarayanan *et al.*, 2017]. The initialization serves as the only source of stochasticity in the model parameters which are otherwise not random; Deep Ensembles can optionally be coupled with a differently shuffled data loader. In contrast to SWAG, results are obtained by averaging the predictions of M independently trained networks instead of explicitly modeling a posterior and sampling from it. [Ovadia *et al.*, 2019] point out that even an ensemble size of $M = 5$ performs well, strengthening its reputation as a "gold standard" for accurate and well-calibrated predictive distributions.

3.3 Uncertainty Decomposition

In the literature two sources of uncertainty are commonly considered [Hüllermeier and Waegeman, 2021]: (1) *Aleatoric* uncertainty represents stochasticity inherent in the data. For the OnHW application this can be sensor noise induced by the ballpoint pen on the paper or by shaky hands of the writer. In particular, even with infinitely many data points, there will always be some variation in the data. (2) *Epistemic* uncertainty is the model uncertainty, which, in theory, can be reduced to zero for an increasing amount of observations. Various approaches of measuring uncertainty exist in the literature. We consider two approaches, both providing justified and mutually complementing insights into our trained models and data situation: uncertainty decomposition based on the softmax output variability [Kwon *et al.*, 2018] in Section 3.3.1 and based on information theory in Section 3.3.2.

3.3.1 Uncertainty Decomposition based on [Kwon et al.]

The definition proposed by [Kwon *et al.*, 2018] is based on considerations by [Kendall and Gal, 2017] and presents a novel way to estimate predictive uncertainty by breaking it down into

$$\underbrace{\frac{1}{T}\sum_{t=1}^{T}\operatorname{diag}(\hat{c}_t) - \hat{c}_t\hat{c}_t^\top}_{\text{aleatoric uncertainty}} + \underbrace{\frac{1}{T}\sum_{t=1}^{T}(\hat{c}_t - \bar{c})(\hat{c}_t - \bar{c})^\top}_{\text{epistemic uncertainty}}, \quad (3)$$

with $\hat{c}_t = (\hat{c}_{t,1}, \ldots, \hat{c}_{t,K}) \in [0,1]^K$ being the softmax output of the NN based on one forward pass (out of T stochastic forward passes), $\sum_{i=1}^{K} \hat{c}_{t,i} = 1$, and $\bar{c} = \frac{1}{T}\sum_{t=1}^{T} \hat{c}_t$.

Interpretation. Equation 3 yields two $K \times K$ matrices with different interpretations. For the *aleatoric* part, diagonal values are in $\{x - x^2 \mid x \in [0,1]\}$, with the maximum uncertainty for $x = 0.5$. If the model is "unsure", meaning that the model neither displays confidence that a prediction corresponds to a certain class nor displays confidence that it is not, we expect high aleatoric uncertainty. The off-diagonal elements consist of values in $\{-x \cdot y \mid x, y \in [0,1]\}$, which yields values on the interval $[-0.25, 0]$. Lower values correspond to higher data uncertainty. For the *epistemic* part, the diagonal contains the squared difference to the mean softmax outputs (over T samples). The off-diagonal has positive values when the softmax values coincide and negative values if the softmax values display an inverse relationship.

3.3.2 Uncertainty Decomposition based on Information Theory

Another way to decompose predictive uncertainty into an aleatoric and epistemic part is by following [Depeweg *et al.*, 2018] and similarly [Smith and Gal, 2018]. Based on principles from information theory, the Shannon entropy $H(p) = -\sum_{i=1}^{K} p_i log_2(p_i)$ is utilized as a common measure of "informedness" of a single probability distribution p with K outcomes/classes and the associated probabilities for each i-th class p_i; taking the logarithm to base 2 yields values measured in *bits*. The total predictive uncertainty (TU) of the predictive distribution $p(y^*|x, D)$ can then be quantified by

$$TU = H\big(p(y^*|x^*, D)\big) \approx H\Big(\frac{1}{S}\sum_{s=1}^{S} p(y^*|x^*, \theta_s)\Big). \quad (4)$$

Effectively, this is the entropy of the averaged categorical predictions, and it includes the two sources of uncertainty we are interested in.

Aleatoric Uncertainty (AU), Entropy. We can express aleatoric uncertainty as the expectation over the entropies of S sampled conditional predictive distributions with fixed weights, i.e.,

$$AU \approx \frac{1}{S}\sum_{s=1}^{S} H(p(y^*|x^*, \theta_s)). \quad (5)$$

Epistemic Uncertainty (EU), Mutual Information. Finally, epistemic uncertainty emerges as the difference of total and aleatoric uncertainty $EU = TU - AU$, and is equivalent to the mutual information (MI):

$$EU = H\Big(\frac{1}{S}\sum_{s=1}^{S} p(y^*|x^*, \theta_s)\Big) - \frac{1}{S}\sum_{s=1}^{S} H(p(y^*|x^*, \theta_s)). \quad (6)$$

Intuitively, epistemic uncertainty stands for the information gain about the model parameters that would be obtained when observing the true outcome. MI is always non-negative, zero in case of perfect independence of y^* and θ, and positive when model uncertainty is present at prediction time.

4 Experiments

In our order to evaluate the efficacy of UQ methods for spatio-temporal handwriting datasets, we use the OnHW dataset (Section 4.1) and fit different network architectures (Section 4.2). Our evaluation approach is given in Section 4.3. For architecture and training details and SWAG parameters, see Appendix A.1. For Deep Ensembles, we choose $M = 10$ (for a study on number of base learners in Deep Ensembles vs. SWAG performance, see [Maddox *et al.*, 2019]).

4.1 Online Handwriting Recognition

The *OnHW-chars* [Ott *et al.*, 2020] dataset consists of recordings from a sensor-enhanced ballpoint pen providing 14 sensor measurements: two accelerometers (3 axes each), one gyroscope (3 axes), one magnetometer (3 axes), a force sensor (with which the pen tip touches the surface), and the time steps. 119 right-handed and nine left-handed writers participated in the data collection. Each person was instructed to write the English alphabet on plain paper six times. This results in 31,275 right-handed and 2,270 left-handed samples. The task is to either classify lowercase letters (26 classes), uppercase letters (26 classes) or combined letters from all 52 classes. For model evaluation, five cross-validation sets are provided by [Ott *et al.*, 2020] for both writer-dependent (WD) and writer-independent (WI) MTS classification tasks.

4.2 Neural Network Architectures

We use a modified CNN from [Ott *et al.*, 2020; 2022d] for feature extraction and combine it with one unit for spatio-temporal classification to extract important temporal features. This unit is added before the last dense layer. We compare a standard long short-term memory (LSTM) cell with 100 neurons, a bidirectional LSTM (BiLSTM) cell with 100 neurons, and a temporal convolutional network (TCN) with 120 neurons. The last dense layer contains 26 neurons for the lowercase and uppercase tasks, or 52 neurons for the combined task. We interpolate the time-series to 64 time steps without sensor normalization.

4.3 Evaluation Metrics

Confidence Calibration. Calibration can be understood as the degree of reliability of a model. According to [Gawlikowski *et al.*, 2021], a predictor is well-calibrated if the derived predictive confidence represents a good approximation of the actual probability of correctness – meaning that 20% of all predictions with a predictive confidence of 80% should actually be false. Calibration is thus a notion of uncertainty, measuring the discrepancy between the model's forecasts and (empirical) long-run frequencies [Lakshminarayanan *et al.*, 2017]. Using the definitions of confidence and accuracy [Guo *et al.*, 2017], we can make statements about over- and under-confidence of the model. We have

$$\text{confidence}(b_e) = \frac{1}{|b_e|} \sum_{s \in b_e} \hat{c}_s \tag{7}$$

and

$$\text{accuracy}(b_e) = \frac{1}{|b_e|} \sum_{s \in b_e} \mathbb{1}(\hat{y}_s = y_s), \tag{8}$$

Method	Lowercase		Uppercase		Combined	
	WD	**WI**	**WD**	**WI**	**WD**	**WI**
Frequentist [Ott *et al.*, 2020]	**84.62** TCN	76.85 TCN	89.89 TCN	**83.01** TCN	70.50 TCN	64.13 LSTM
SWAG	84.44 TCN	**76.96** TCN	87.58 TCN	82.21 TCN	72.54 TCN	**66.12** TCN
Deep Ensembles	83.43 BiLSTM	73.41 TCN	**90.31** TCN	81.26 TCN	**75.51** TCN	64.21 TCN

Table 1: Accuracies (in %) for models trained on *right-handed* writers data and evaluated on *right-handed* writers data. Second row shows the respective model. **Bold**: best results.

with b_e denoting the set of indices of sampled softmax outputs falling into the interval $(l_e, u_e]$. Commonly, the softmax output range is divided into ten bins (interval sizes of 0.1). We can now make statements whether our model is under-confident $\big(\text{accuracy}(b_e) > \text{confidence}(b_e)\big)$ or over-confident $\big(\text{accuracy}(b_e) < \text{confidence}(b_e)\big)$. It has been shown that softmax outputs of deep NNs are in general not well calibrated and are often either over- or under-confident [Guo *et al.*, 2017]. Ideally, $\text{accuracy}(b_e) \approx \text{confidence}(b_e)$, allowing the user to interpret softmax outputs as probabilities and thereby quantify the prediction uncertainty.

Expected Calibration Error (ECE). The ECE summarizes how far away the confidence is from the actual (empirical) accuracy [Guo *et al.*, 2017]. It can be defined as

$$\text{ECE}(b_e) = \sum_{e=1}^{E} \frac{|b_e|}{n} |\text{accuracy}(b_e) - \text{confidence}(b_e)|, \tag{9}$$

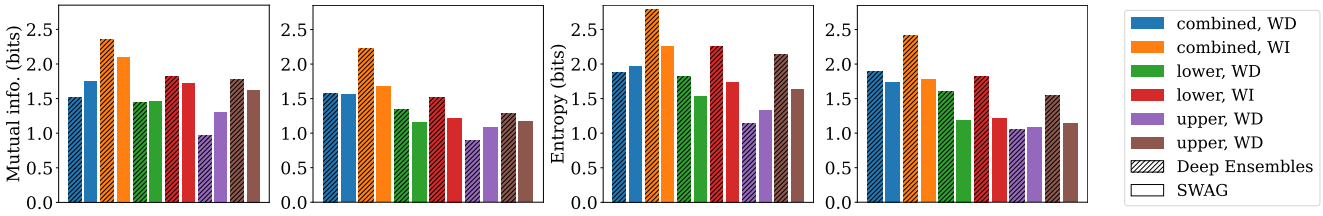
with n being the number of predicted softmax outputs, and E being the number of bins. Note that this metric does not give any information about over- or under-confidence – only how far away the expected accuracy is from the confidence. Ideally, the ECE is 0.

Reliability Diagrams. We visualize Equations 7 and 8 as reliability diagrams [Degroot and Fienberg, 1983] for selected models. Generally, a model is over-confident if the black bars (displaying the accuracy for one bin) are below the dashed bisectors. Consequently, if the black bars are above the bisectors, the model is under-confident. We additionally plot the histogram [Hollemans, 2020] of the softmax outputs to get an overview of the distribution.

5 Experimental Results

In the following, we summarize the main results. In general, the models perform better on WD classification tasks than on WI tasks. Architectures with TCN units outperform LSTM and BiLSTM units on most tasks.

Evaluation on Handedness (trained on right-handed writers). SWAG and Deep Ensemble models perform very similarly to frequentist models proposed in [Ott *et al.*, 2020] in terms of predicitve accuracy (see Table 1), being at most 3% points below and 5% points above a respective frequentist model. When applying models trained with right-handed data on the left-handed datasets, the performance ranges from



(a) Mutual information. (b) Entropy. (c) Legend.

Figure 1: Information theoretic uncertainty measures for the Deep Ensemble (dashed) and SWAG (non-dashed) CNN+TCN models. The models are trained on the combined right- and left-handed writers datasets (a and b, left) or only the right-handed writers dataset (a and b, right), and evaluated on the left-handed writers dataset. We provide results for lowercase, uppercase and combined classification tasks.

Method		Lowercase		Uppercase		Combined	
		WD	WI	WD	WI	WD	WI
SWAG	right	83.73	76.27	87.10	81.69	72.13	65.41
	left	**55.51**	**45.91**	55.04	**50.67**	**46.08**	**39.26**
Deep Ensembles	right	83.07	73.87	89.92	80.86	75.29	64.22
	left	45.25	37.00	**62.73**	48.31	45.95	33.27
Best BNN Method (right-handed)	right	**84.44**	**76.96**	**90.31**	**82.21**	**75.51**	**66.12**
	left	42.55	44.19	49.87	48.54	33.68	36.20

Table 2: Accuracies (in %) for best models trained on right- and left-handed data and evaluated on *right-handed or left-handed* writers data *separately*, compared to the best performing models which were only trained on right-handed data. **Bold**: best results.

Method	Lowercase		Uppercase		Combined	
	WD	WI	WD	WI	WD	WI
SWAG	**81.85**	**74.24**	84.92	**79.58**	70.37	**63.64**
	TCN	TCN	TCN	TCN	TCN	TCN
Deep Ensembles	80.55	71.41	**88.07**	78.65	**73.31**	62.14
	LSTM	TCN	TCN	TCN	TCN	TCN

Table 3: Accuracies (in %) for models trained on *right- and left-handed* writers data and evaluated on *right-handed* writers data. Second row shows the respective model. **Bold**: best results.

33.27% to 49.87% accuracy (see Table 2) which is substantially better than "pure guessing" – our models make informed decisions after shifting domains, albeit at a lower standard. A possible reason is that certain sensors produce nearly identical signals regardless of the orientation of the pen. For example, the accelerometer at the bottom of the pen should give the same readings for left-handed writers when writing `"I"` and `"i"` as for right-handed writers, since it is simply a downward motion regardless of the writing hand.

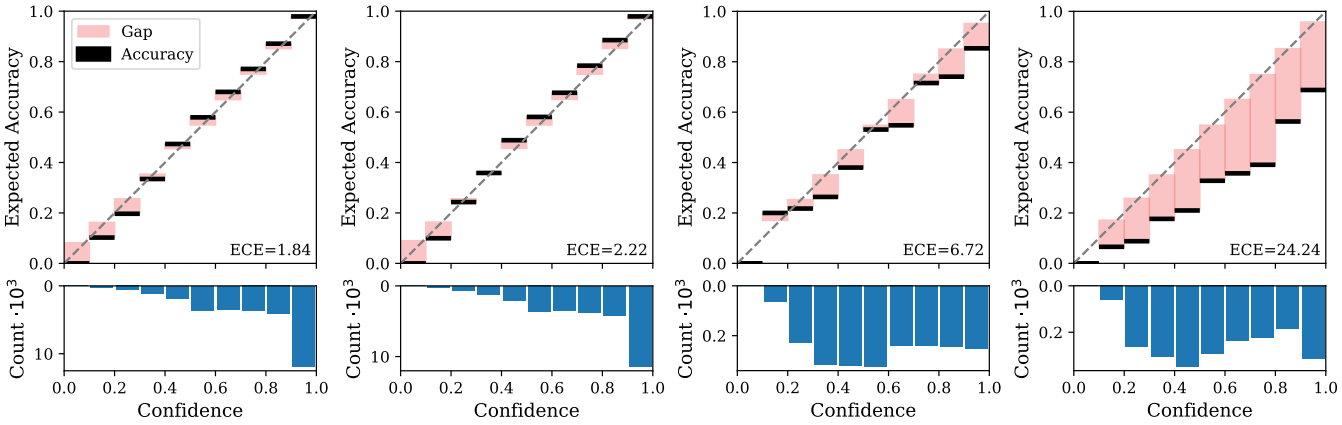
Evaluation on Handedness (trained on right- and left-handed writers). When evaluating performance on right-handed data, models trained only on right-handed datasets consistently outperform models trained on both datasets combined and yield between 2% points and 12% points higher accuracies (see Table 3). This performance loss is compensated by a performance gain for left-handed data. Still, the performance is not up to par with right-handed data; this gap may be due to a "writing style" particular to every writer that especially influences the gyroscope and magnetometer measurements. More importantly, left-handed writers have a writing style different to right-handed writers which is perhaps exactly what the right-handed models never learned in order to address the style of left-handed writers, underlining the need for a sufficient amount of samples to get a good representation of various writing styles.

Analysis of Uncertainty. Figure 1 shows the MI and entropy for SWAG and Deep Ensemble models evaluated on the left-handed data. The barplots show that the models trained on only right-handed data display lower uncertainty (i.e., higher confidence) compared to models trained on combined data. However, this higher confidence is not empirically justified when looking at the reliability diagrams in Figure 2, which point out that models trained without left-handed writers data are miscalibrated and therefore overconfident. Models trained on the combined writers (Figures 2a and 2c) provide more realistic accuracies when applied to the left-handed data (ECE of 6.72). The ECE is even higher (24.24) for left-handed evaluation without left-handers in the training set (see Figure 2d). For a separate evaluation for each character, see Appendix A.2.

5.1 Uncertainty Analysis based on [Kwon et al.]

In Figure 3 we visualize the aleatoric and epistemic uncertainty as well as the confusion matrix for the Deep Ensemble model and the combined task. For SWAG model results, see Appendix A.3. In the aleatoric uncertainty heatmap (Figure 3a) we observe a trace with negative values at the lower end of the scale. Note that for off-diagonal values, the aleatoric uncertainty is higher for lower softmax values. Here, two softmax outputs (with the highest values) coincide on average (see Section 3.3.1). This means that the model tends to confuse the two classes. The most prominent off-diagonal strip corresponds to the upper- and lowercase pairs. This makes intuitive sense since, e.g., the lowercase `"u"` and uppercase `"U"` are written similarly. This effect is consequently not present for less similar pairs like `"a"` and `"A"`. We can see this effect also for `"l"` (lowercase `"L"`) and `"I"` (uppercase `"i"`). A very similar pattern can also be observed in the confusion matrix (see Figure 3c), confirming that the trained model is not only unsure about how to classify these pairs, but is also empirically worse in the respective classification task.

These patterns allow for further interesting insights. For example, one might expect this pattern to occur for `"i"` and `"j"`, but the corresponding heatmap entries lack signs of confusion of the model. Similarity between characters consequently hinges on the similarity of *motions* while writing.



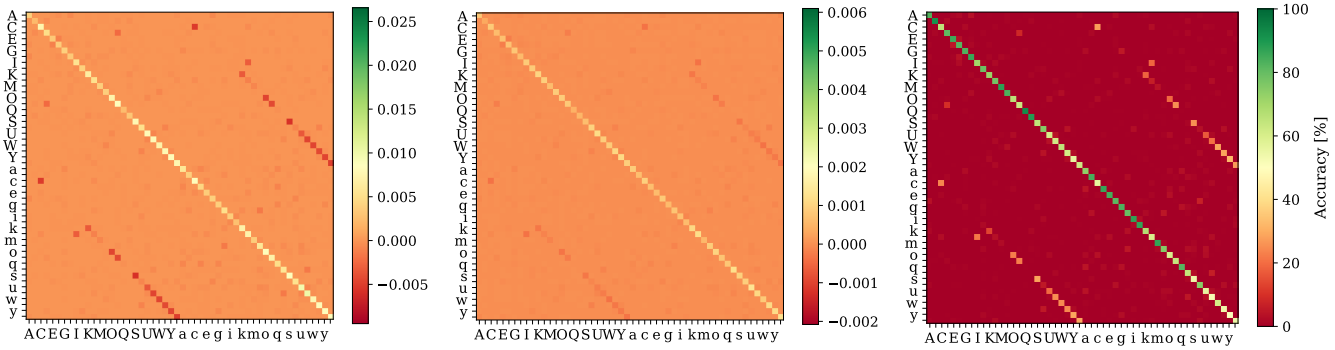
(a) Evaluated on right-handed writers data.

(b) Evaluated on right-handed writers data.

(c) Evaluated on left-handed writers data.

(d) Evaluated on left-handed writers data.

Figure 2: Reliability diagram for the Deep Ensemble CNN+TCN model trained on the combined WD datasets. a) and c): Trained on the combined right- and left-handed writers datasets. b) and d): Trained on right-handed writers only.

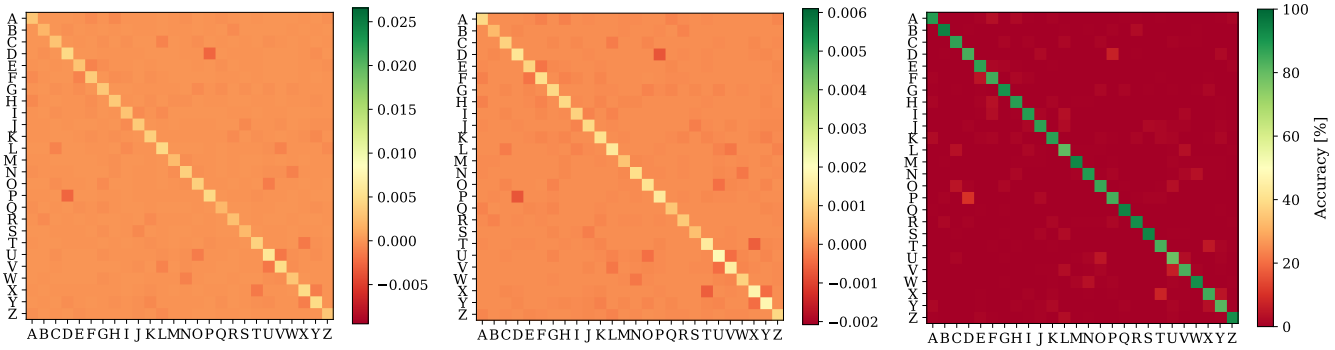


(a) Aleatoric uncertainty.

(b) Epistemic uncertainty.

(c) Confusion matrix of accuracy.

Figure 3: Uncertainty prediction for the Deep Ensemble CNN+TCN model trained on the combined WD (right-handed only) dataset. Note that the color scale is fixed for all subplots for comparability with Figure 4, and that we skipped every second character label for readability.



(a) Aleatoric uncertainty.

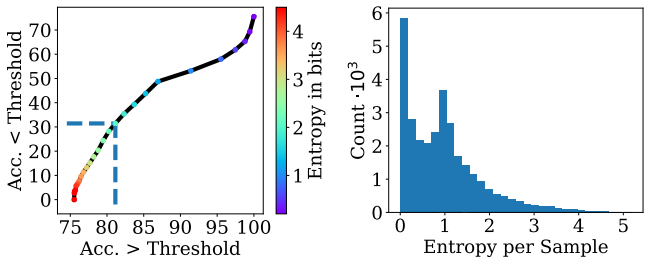
(b) Epistemic uncertainty.

(c) Confusion matrix of accuracy.

Figure 4: Uncertainty prediction for the Deep Ensemble CNN+TCN model trained on the uppercase WD (right-handed only) dataset. Note that the color scale is fixed for all subplots for comparability with Figure 3 and 6.

Two characters with small differences are written similarly but in different *sizes*. This also holds for specific parts of the letters. For example, "n" and "h" have a higher aleatoric uncertainty in Figure 3a; the major difference being that one tiny part of "h" is longer.

Somewhat puzzling is that we see the same effect in the epistemic uncertainty heatmap (see Figure 3b), where such pairs with high similarity lead to negative values. When one entry of the softmax output values is below and another entry above the respective sample mean, negative epistemic uncertainty is implied. This leads to some kind of discriminative power due to the negative "covariance" for which there is lit-



(a) Sample accuracies below and above an entropy threshold.

(b) Histogram visualizing the entropy distribution.

Figure 5: Accuracy and entropy for the Deep Ensemble CNN+TCN model trained on the combined WD (right-handed only) dataset.

tle justification. We thus advise caution when interpreting the epistemic uncertainty in this context.

5.2 Uncertainty based on Information Theory

We further highlight the trade-off when using information theory-based measures to decide whether a sample is too uncertain to classify correctly. This is depicted by Figure 5a showing the relationship between classification accuracies and different threshold values. We choose the entropy as the target metric for uncertainty evaluation (MI would work analogously). On the x-axis is the accuracy of the samples above the threshold, i.e., samples our model feels confident about classifying correctly. On the y-axis is the accuracy for the samples below the threshold. These values would be considered as too inaccurate to confidently classify. Setting the threshold to 2.0 bits would approximately yield an accuracy of 82% for the observations above this threshold and approx. 31% accuracy for observations below this threshold (emphasized by the dashed lines). Figure 5b depicts the entropy distribution and further clarifies this point. Convincingly, the accuracy reduces to almost zero for very high entropy samples. Note that the accuracy does not need to decrease with an increasing entropy threshold or even be zero for very high entropy values, even though this is generally true for our models.

5.3 Summary and Limitations

Uncertainty Decomposition. Neither uncertainty quantification method shows notable differences between aleatoric and epistemic uncertainty. The heatmaps exhibit the same "strip" for similar characters and give no hints to different sources of uncertainty (data-driven or systemic confusion). The benefits of this kind of uncertainty differentiation are limited, but measuring the total uncertainty can still be useful for domain adaptation or the detection of wrong labels.

Real-World Link. Since the models trained on right- and left-handed writers lead to lower data confidence compared to models trained only on right-handed writers (see Figure 1), it is unclear how well the measured MI and entropy translate to the real-world uncertainty. Therefore, verifying uncertainty remains a limitation in our interpretation. While we can discriminate between the entropy associated with different samples, pre-defining thresholds for uncertain samples is challenging due to the following reasons: (1) Raw sensor data is elaborate to interpret and making statements about, e.g., the writing style from sensor data is hardly possible – which, in turn, is connected to model uncertainty. (2) Interpreting the *graphomotoricity* qualitatively, e.g., for teaching hand writing, a qualified expert in this field is required. (3) Different writing domains (different pens, surfaces etc.) lead to different requirements for the uncertainty threshold.

6 Conclusion

We employed SWAG and Deep Ensembles for OnHW recognition with left- and right-handed writers, a spatio-temporal MTS classification task with domain shift. We critically evaluated aleatoric and epistemic uncertainty using confidence calibration, ECE and reliability diagrams. In summary, (1) the model performance only partly relates to the handedness of writers, (2) our models are over-confident and miscalibrated when only trained with right-handed writers and evaluated on left-handed writers, (3) the uncertainty of the models for small and capital characters combined is related to lower classification accuracy, and (4) the entropy and mutual information for individual samples correlate well with the accuracy of our models. Our comparison of different ways to decompose uncertainty easily generalizes to other classification tasks and can be useful for spatio-temporal reasoning. In terms of Bayesian inference, SWAG and Deep Ensemble models perform similarly, while SWAG is computationally less expensive.

Acknowledgements

This work was supported by the Federal Ministry of Education and Research (BMBF) of Germany by Grant No. 01IS18036A (David Rügamer) and by the research program Human-Computer-Interaction through the project "Schreibtrainer", Grant No. 16SV8228, as well as by the Bavarian Ministry for Economic Affairs, Infrastructure, Transport and Technology through the Center for Analytics-Data-Applications (ADA-Center) within the framework of "BAYERN DIGITAL II".

References

[Blundell *et al.*, 2015] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight Uncertainty in Neural Network. In *ICML*, volume 37, pages 1613–1622, July 2015.

[Cai *et al.*, 2014] Ruichu Cai, Jiawei Chen, Zijian Li, Wei Chen, Keli Zhang, Junjian Ye, Zhuozhang Li, Xiaoyan Yang, and Zhenjie Zhang. Time Series Domain Adaptation via Sparse Associative Structure Alignment. In *AAAI*, volume 216, pages 76–102, 2014.

[Daxberger *et al.*, 2021] Erik Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, Matthias Bauer, and Philipp Hennig. Laplace Redux - Effortless Bayesian Deep Learning. In *NIPS*, December 2021.

[Degroot and Fienberg, 1983] Morris H. Degroot and Stephen E. Fienberg. The Comparison and Evaluation of Forecasters. In *The Statistician*, volume 32, 1983.

[Depeweg *et al.*, 2018] Stefan Depeweg, Jose-Miguel Hernandez-Lobato, Finale Doshi-Velez, and Steffen Udluft. Decomposition of Uncertainty in Bayesian Deep Learning for Efficient and

Risk-sensitive Learning. In *JMLR*, volume 80, pages 1184–1193, 2018.

[Gawlikowski *et al.*, 2021] Jakob Gawlikowski, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, Muhammad Shahzad, Wen Yang, Richard Bamler, and Xiao Xiang Zhu. A Survey of Uncertainty in Deep Neural Networks. In *arXiv:2107.03342*, July 2021.

[Gómez *et al.*, 2021] Jairo Alejandro Gómez, ChengHe Guan, Pratyush Tripathy, Juan Carlos Duque, Santiago Passos, Michael Keith, and Jialin Liu. Analyzing the Spatiotemporal Uncertainty in Urbanization Predictions. In *Remote Sensing*, volume 13(512), 2021.

[Guo *et al.*, 2017] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On Calibration of Modern Neural Networks. In *ICML*, volume 70, pages 1321–1330, August 2017.

[Hoeting *et al.*, 1999] Jennifer A. Hoeting, David Madigan, Adrian E. Raftery, and Chris T. Volinsky. Bayesian Model Averaging: A Tutorial. In *Statist. Sci.*, volume 14(4), pages 382–417, November 1999.

[Hollemans, 2020] Matthijs Hollemans. Reliability Diagrams. https://github.com/hollance/reliability-diagrams, 2020.

[Hüllermeier and Waegeman, 2021] Eyke Hüllermeier and Willem Waegeman. Aleatoric and Epistemic Uncertainty in Machine Learning: An Introduction to Concepts and Methods. In *Machine Learning*, volume 110, pages 457–506, March 2021.

[Izmailov *et al.*, 2018] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging Weights Leads to Wider Optima and Better Generalization. In *UAI*, 2018.

[Kendall and Gal, 2017] Alex Kendall and Yarin Gal. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision. In *NIPS*, volume 30, pages 5580–5590, December 2017.

[Khan *et al.*, 2018] Mohammad Khan, Didrik Nielsen, Voot Tangkaratt, Wu Lin, Yarin Gal, and Akash Srivastava. Fast and Scalable Bayesian Deep Learning by Weight-Perturbation in Adam. In *JMLR*, volume 80, pages 2611–2620, 2018.

[Kwon *et al.*, 2018] Yongchan Kwon, Joong-Ho Won, Beom Joan Kim, and Myunghee Cho Paik. Uncertainty Quantification Using Bayesian Neural Networks in Classification: Application to Ischemic Stroke Lesion Segmentation. In *ICLR*, 2018.

[Lakshminarayanan *et al.*, 2017] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and Scalable Predictive Uncertainty Estimation Using Deep Ensembles. In *NIPS*, pages 6405–6416, December 2017.

[Li *et al.*, 2022] Xiaotong Li, Yongxing Dai, Yixiao Ge, Jun Liu, Ying Shan, and Lingyu Duan. Uncertainty Modeling for Out-of-Distribution Generalization. In *ICLR*, 2022.

[Long *et al.*, 2014] Mingsheng Long, Jianmin Wang, Guiguang Ding, Jiaguang Sun, and Philip S. Yu. Transfer Joint Matching for Unsupervised Domain Adaptation. In *CVPR*, pages 1410–1417, Columbus, OH, June 2014.

[Maddox *et al.*, 2019] Wesley J. Maddox, Timur Garipov, Pavel Izmailov, Dmitry Vetrov, and Andrew Gordon Wilson. A Simple Baseline for Bayesian Uncertainty in Deep Learning. In *NIPS*, pages 13153–13164, December 2019.

[Ott *et al.*, 2020] Felix Ott, Mohamad Wehbi, Tim Hamann, Jens Barth, Björn Eskofier, and Christopher Mutschler. The OnHW Dataset: Online Handwriting Recognition from IMU-Enhanced Ballpoint Pens with Machine Learning. In *IMWUT*, volume 4(3), article 92, Cancún, Mexico, September 2020.

[Ott *et al.*, 2022a] Felix Ott, David Rügamer, Lucas Heublein, Bernd Bischl, and Christopher Mutschler. Cross-Modal Common Representation Learning with Triplet Loss Functions. In *arXiv:2202.07901*, February 2022.

[Ott *et al.*, 2022b] Felix Ott, David Rügamer, Lucas Heublein, Bernd Bischl, and Christopher Mutschler. Domain Adaptation for Time-Series Classification to Mitigate Covariate Shift. In *arXiv:2204.03342*, April 2022.

[Ott *et al.*, 2022c] Felix Ott, David Rügamer, Lucas Heublein, Bernd Bischl, and Christopher Mutschler. Joint Classification and Trajectory Regression of Online Handwriting using a Multi-Task Learning Approach. In *WACV*, pages 266–276, Waikoloa, HI, January 2022.

[Ott *et al.*, 2022d] Felix Ott, David Rügamer, Lucas Heublein, Tim Hamann, Jens Barth, Bernd Bischl, and Christopher Mutschler. Benchmarking Online Sequence-to-Sequence and Character-based Handwriting Recognition from IMU-Enhanced Pens. In *arXiv:2202.07036*, February 2022.

[Ovadia *et al.*, 2019] Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua V. Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift. In *NIPS*, volume 32, pages 14003–14014, December 2019.

[Pan and Yang, 2009] Sinno Jialin Pan and Qiang Yang. A Survey on Transfer Learning. In *Trans. on Knowledge and Data Engineering*, volume 22(10), pages 1345–1359, October 2009.

[Plamondon and Srihari, 2000] Rejean Plamondon and Sargur N. Srihari. On-line and Off-line Handwriting Recognition: A Comprehensive Survey. In *TPAMI*, volume 22(1), pages 63–84, January 2000.

[Saenko *et al.*, 2010] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting Visual Category Models to New Domains. In *ECCV*, volume 6314, pages 213–226, 2010.

[Schölkopf *et al.*, 2021] Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward Causal Representation Learning. In *Proceedings of the IEEE*, volume 109(5), pages 612–634, 2021.

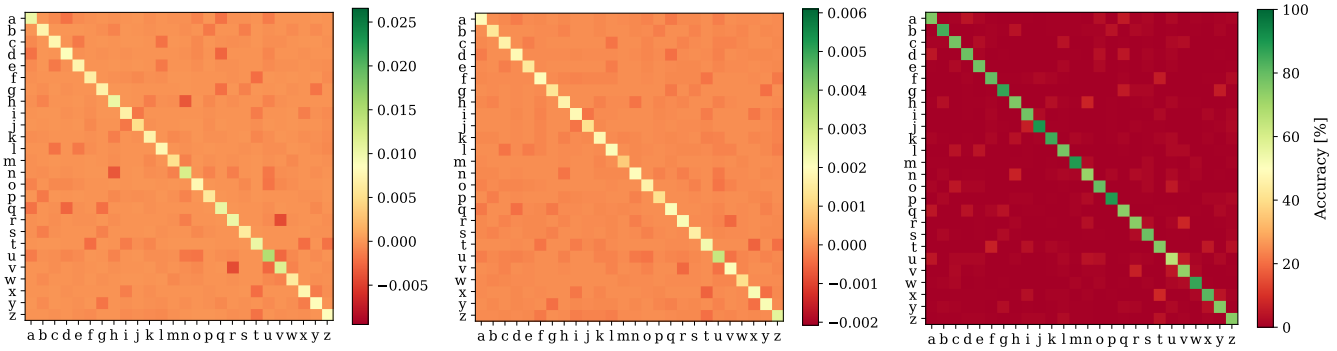
[Shao *et al.*, 2014] Ling Shao, Fan Zhu, and Xuelong Li. Transfer Learning for Visual Categorization: A Survey. In *Trans. on Neural Networks and Learning Systems*, volume 26(5), pages 1019–1034, July 2014.

[Smith and Gal, 2018] Lewis Smith and Yarin Gal. Understanding Measures of Uncertainty for Adversarial Example Detection. In *UAI*, 2018.

[Sun *et al.*, 2016] Baochen Sun, Jiashin Feng, and Kate Saenko. Correlation Alignment for Unsupervised Domain Adaptation. In *arXiv:1612.01939*, December 2016.

[Wu *et al.*, 2021] Dongxia Wu, Liyao Gao, Xinyue Xiong, Matteo Chinazzi, Alessandro Vespignani, Yi-An Ma, and Rose Yu. Quantifying Uncertainty in Deep Spatiotemporal Forecasting. In *arXiv:2105.11982*, May 2021.

[Zhou *et al.*, 2021] Zhengyang Zhou, Yang Wang, Xike Xie, Lei Qiao, and Yuantao Li. STUaNet: Understanding Uncertainty in Spatiotemporal Collective Human Mobility. In *WWW*, pages 1868–1879, April 2021.



(a) Aleatoric uncertainty. (b) Epistemic uncertainty. (c) Confusion matrix of accuracy.

Figure 6: Uncertainty prediction for the Deep Ensemble CNN+BiLSTM model (which outperformed the TCN-based architecture) trained on the lowercase WD (right-handed only) dataset. Note that the color scale is fixed for all subplots for comparability with Figure 3 and 4.

A Appendices

We propose model parameters in Section A.1 and show an evaluation per character in Section A.2. We propose results for the SWAG model in Section A.3.

A.1 Model and UQ Method Parameters

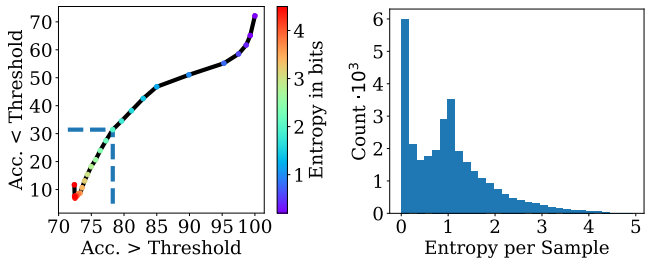
For reproducibility, we state all general model architecture parameters and propose training parameters for the SWAG model. For all experiments we use Nvidia Tesla V100-SXM2 GPUs with 32 GB VRAM coupled with Intel Core Xeon CPUs and 192 GB RAM.

Model Parameters. We use a CNN with dropout rate 20%, convolutional layers with kernel size 4 and filter size 200. The temporal cell (LSTM, BiLSTM or TCN) contains 100, 100 or 120 neurons, respectively. We interpolate the time-series to 64 time steps, and train the model for 2,000 epochs with early stopping and a batch size of 50.

SWAG Parameters. We initialize the stochastic gradient descent (SGD) optimizer with initial learning rate 10^{-2}, a momentum of 0.9, and weight decay of 10^{-4}. The stochastic weight averaging (SWA) burn-in period was run for 10 epochs. SWAG showed a training process with fast convergence.

A.2 Evaluation per Character

Confusion Matrices. We propose the confusion matrices for the aleatoric and epistemic uncertainty as well as the accuracy (in %) for the uppercase (see Figure 4) and lowercase (see Figure 6) datasets. While for the combined training, lower- and uppercase characters are often misclassified, the separate training leads to confusion of characters with similar shapes, e.g., for the uppercase task, the model is uncertain for "D" and "P", "U" and "V", and "T" and "X". These confusions can be identified with the aleatoric and epistemic uncertainty and correspond with the classification accuracies. Overall, the uncertainty for lowercase characters is higher (see Figure 6a) since the writing style of lowercase characters is oftentimes quite similar, e.g., "r" and "v", "u" and "v", "h" and "n", and "d" and "q". This also leads to a lower classification accuracy (see Figure 6c).



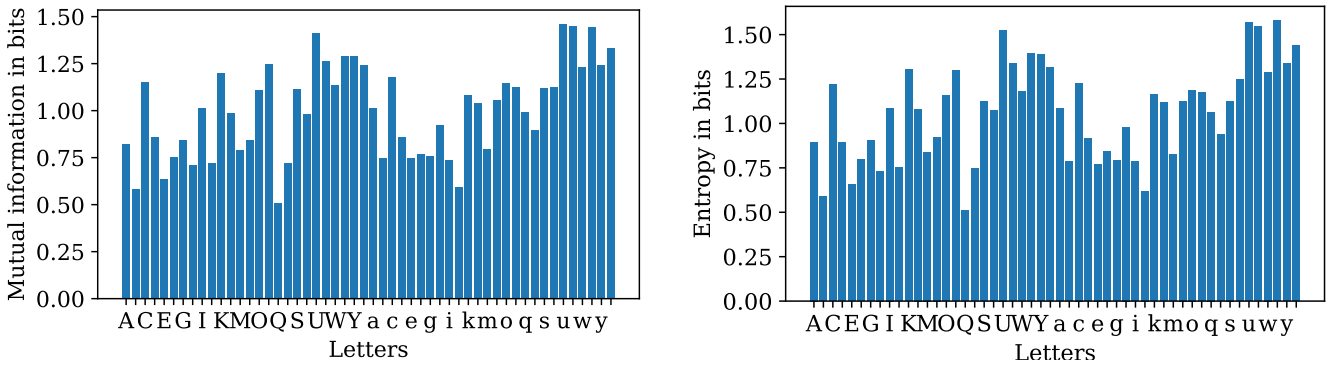
(a) Sample accuracies below and above an entropy threshold. (b) Histogram visualizing the entropy distribution.

Figure 7: Accuracy and entropy for the SWAG CNN+TCN model trained on the combined WD (right-handed only) dataset.

Mutual Information and Entropy. Figure 8a shows the mutual information (MI) per character, and Figure 8b shows the entropy, respectively. In general, the MI and entropy correlates and are similar for each character. The MI and entropy is high for the characters "U", "u", "v", "x", and "z". Furthermore, both metrics are higher for lowercase characters than for uppercase characters. This corresponds to the confusion matrices in Figure 4 and 6 where aleatoric uncertainty is higher for off-diagonals for lowercase characters.

A.3 SWAG Model Results

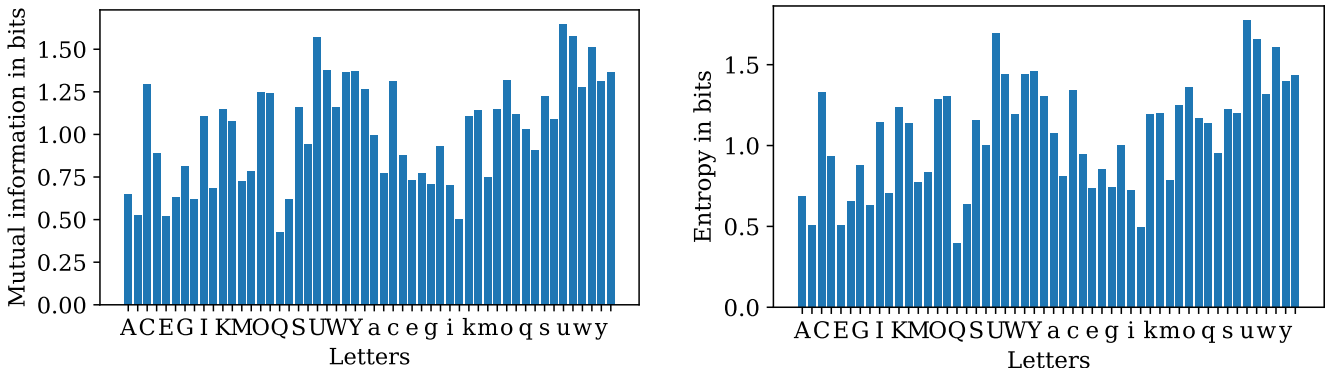
This section provides plots for the SWAG model that can directly be compared to the previously shown Deep Ensemble model plots. We observe very similar results between SWAG and Deep Ensemble models. Figure 9 shows the MI and entropy for the SWAG model with the same pattern as for the Deep Ensemble model with lower absolute values. In Figure 10, we see the same overconfidence on left-handed data for SWAG models that have never seen this data similar as for Deep Ensemble models. The ECE by the SWAG model is marginally lower than the ECE by the Deep Ensemble model, but follows the same trend. The heatmaps in Figures 11 for lowercase and uppercase characters, in Figure 12 for uppercase characters only, and in Figure 13 for lowercase characters only of the SWAG model show the same pattern as the heatmaps for Deep Ensemble models.



(a) Mutual information per letter.

(b) Entropy per letter.

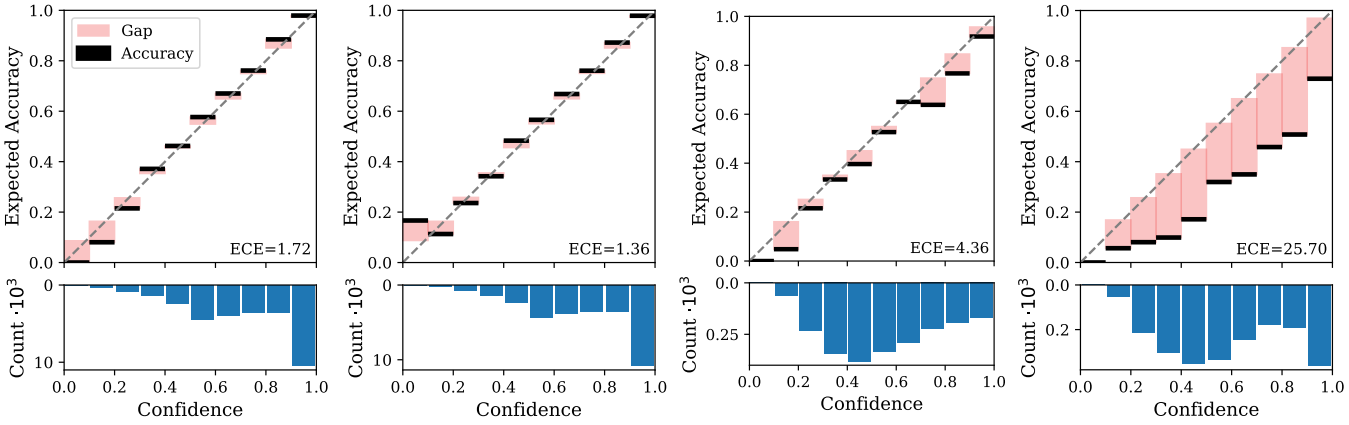
Figure 8: Mutual information and entropy per letter for the Deep Ensemble CNN+TCN model trained on the combined WD (right-handed only) dataset. Note that we skiped every second character in the x-axis (ordered alphabetically) for readability.



(a) Mutual information per letter.

(b) Entropy per letter.

Figure 9: Mutual information and entropy per letter for the SWAG CNN+TCN model trained on the combined WD (right-handed only) dataset. Note that we skipped every second character in the x-axis (ordered alphabetically) for readability.



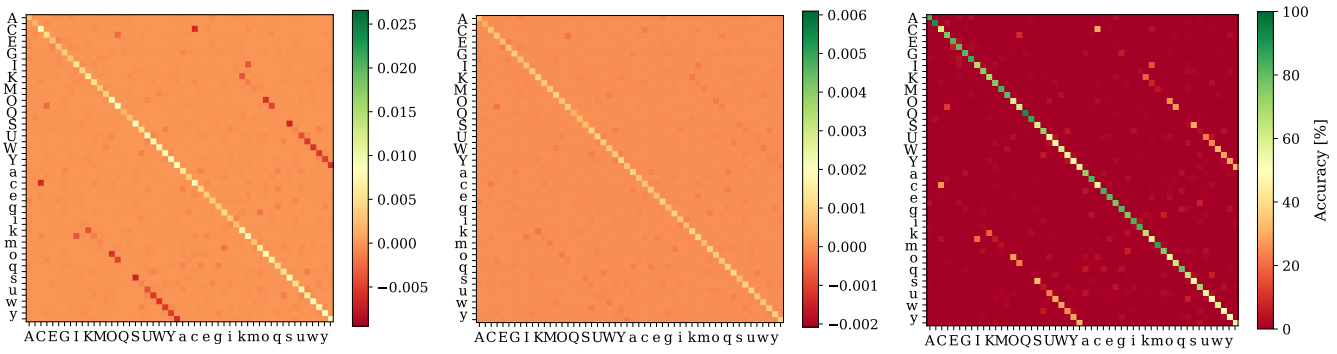
(a) Evaluated on right-handed writers data.

(b) Evaluated on right-handed writers data.

(c) Evaluated on left-handed writers data.

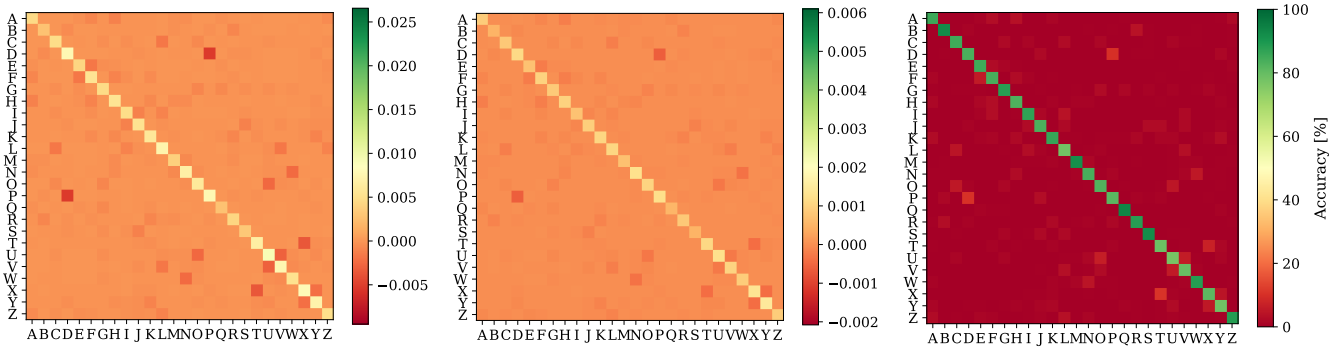
(d) Evaluated on left-handed writers data.

Figure 10: Reliability diagram for the SWAG CNN+TCN model trained on the combined WD datasets. a) and c): Trained on the combined right- and left-handed writers datasets. b) and d): Trained on right-handed writers only.



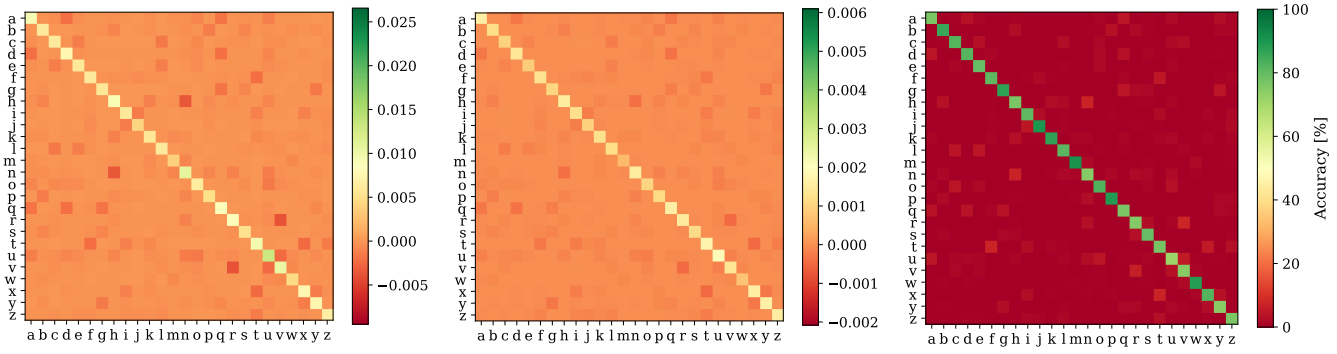
(a) Aleatoric uncertainty. (b) Epistemic uncertainty. (c) Confusion matrix of accuracy.

Figure 11: Uncertainty prediction for the SWAG CNN+TCN model trained on the combined WD (right-handed only) dataset. Note that the color scale is fixed for all subplots for comparability with the other heatmaps, and that we skipped every second character label for readability.



(a) Aleatoric uncertainty. (b) Epistemic uncertainty. (c) Confusion matrix of accuracy.

Figure 12: Uncertainty prediction for the SWAG CNN+TCN model trained on the uppercase WD (right-handed only) dataset. Note that the color scale is fixed for all subplots for comparability with the other heatmaps.



(a) Aleatoric uncertainty. (b) Epistemic uncertainty. (c) Confusion matrix of accuracy.

Figure 13: Uncertainty prediction for the SWAG CNN+TCN model trained on the lowercase WD (right-handed only) dataset. Note that the color scale is fixed for all subplots for comparability with the other heatmaps.