Better Uncertainty Quantification for Machine Translation Evaluation

Anonymous ACL submission

Abstract

Neural-based machine translation (MT) evaluation metrics are progressing fast. However, they are often hard to interpret and might produce unreliable scores when human references or assessments are noisy or when data is out-of-domain. Recent work leveraged uncertainty quantification techniques such as Monte Carlo dropout and deep ensembles to provide confidence intervals, but these techniques (as we show) are limited in several ways. In this paper, we introduce more powerful and efficient uncertainty predictors for capturing both aleatoric and epistemic uncertainty, by training the COMET metric with new heteroscedastic regression, divergence minimization, and direct uncertainty prediction objectives. Our experiments show improved results on WMT20 and WMT21 metrics task datasets and a substantial reduction in computational costs. Moreover, they demonstrate the ability of our predictors to identify low quality references and to reveal model uncertainty due to out-of-domain data.

1 Introduction

Trainable neural-based MT evaluation metrics, such as COMET or BLEURT (Rei et al., 2020a; Sellam et al., 2020a), are becoming increasingly successful (Freitag et al., 2021b). For system comparison, they surpass or complement traditional lexical metrics such as BLEU (Papineni et al., 2002), and at a segment level, they show higher correlations with human judgments, with and without access to references (Kepler et al., 2019; Thompson and Post, 2020; Ranasinghe et al., 2020).

However, MT evaluation metrics need a measure of **confidence** over their quality predictions, so that they can be better contextualized and interpreted. Indeed, neural-based MT evaluation models are prone to multiple sources of epistemic and aleatoric uncertainty, often over- or under-estimating MT quality, specially when applied to new domains or languages. Recently, Glushkova et al. (2021) proposed **uncertainty-aware MT evaluation** by combining COMET with two simple uncertainty quantification methods based on model variance, Monte Carlo (MC) dropout (Gal and Ghahramani, 2016) and deep ensembles (Lakshminarayanan et al., 2017). However, these two methods have two important shortcomings:

- They are costly in terms of inference time (MC dropout) or training time (deep ensembles).
- They are not able to distinguish between different sources of uncertainty. For example, it is impossible to infer whether the uncertainty stems from a noisy and ambiguous reference, an out-of-distribution example, or noisy annotations. More fundamentally, they are highly model-dependent and cannot distinguish between aleatoric and epistemic uncertainty (as illustrated in Figs. 1–2).

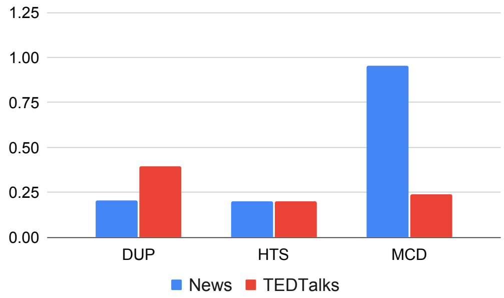


Figure 1: **Epistemic uncertainty caused by out-of-domain data.** We show sharpness (average uncertainty) on two English-German test sets from the WMT21 metrics task: an in-domain dataset (News) and an out-of-domain dataset (TED talks). Our proposed method that handles epistemic uncertainty (DUP) exhibits higher uncertainty on the out-of-domain dataset, as expected. HTS, which detects aleatoric, but not epistemic uncertainty, has similar uncertainty in both datasets, and the MCD baseline, surprisingly, has the opposite behavior.

In this paper, we address the limitation above by investigating more powerful (and efficient) uncertainty quantification methods: **direct uncertainty**

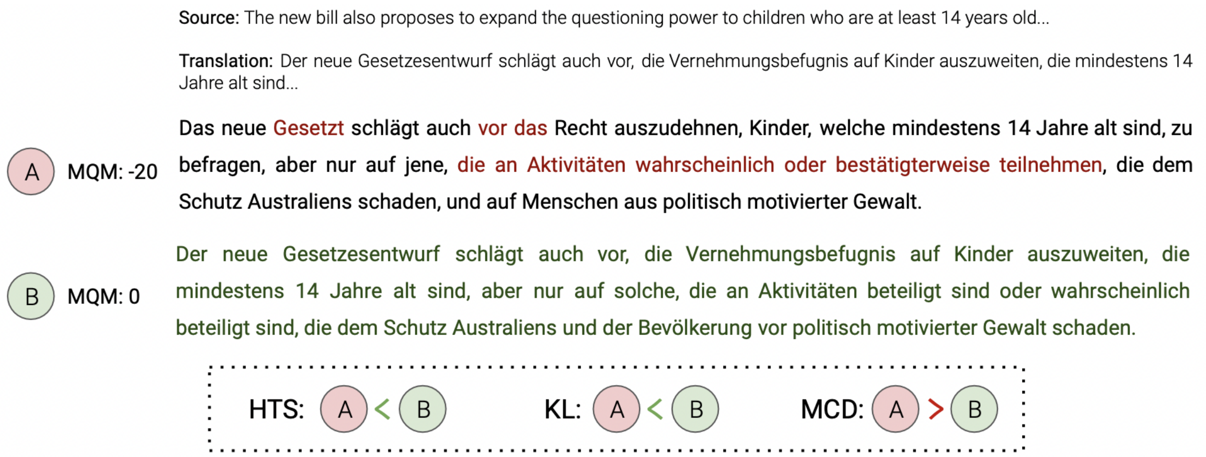


Figure 2: **Aleatoric uncertainty caused by noisy references.** We show a low quality reference (A) and a high quality reference (B) for an English-German translation. Errors in reference A are annotated in dark red; reference B has a perfect MQM score of 0 (no errors). Our two proposed methods that handle aleatoric (data) uncertainty, HTS and KL, are more uncertain when given the low-quality reference, as expected. The previously proposed MCD method (Glushkova et al., 2021) behaves in the opposite way. Full dataset statistics are shown in Figure 4.

prediction (Jain et al., 2021), a two-step approach which uses supervision over the quality prediction errors; **heteroscedastic regression**, which estimates input-dependent aleatoric uncertainty and can be combined with MC dropout (Kendall and Gal, 2017); and **divergence minimization**, which can estimate aleatoric uncertainty from annotator disagreements, when multiple annotations are available for the same example.

We evaluate our newly proposed uncertainty estimators on 16 language pairs from the WMT20 and WMT21 metrics shared task, using two types of human annotations: direct assessments (DA) and multi-dimensional quality metric scores (MQM). The experiments show that our estimators compare favourably against model variance baselines, while being considerably faster. We also show that, contrarily to the baselines, our proposed methods are effective at detecting potentially incorrect references and out-of-distribution examples in the data.[1]

2 Related Work

MT evaluation Traditional metrics for MT evaluation, including BLEU (Papineni et al., 2002), METEOR (Lavie and Denkowski, 2009), and CHRF (Popović, 2015) are based on lexical overlap. More recent metrics leverage large pretrained models, both unsupervised, such as BERTSCORE (Zhang et al., 2019), YISI (Lo, 2019) and PRISM (Thompson and Post, 2020), or fine-tuned on human annotations, such as COMET (Rei et al., 2020a) and BLEURT (Sellam et al., 2020b). In recent studies it has become increasingly evident that supervised metrics exhibit higher correlations with human judgements (Mathur et al., 2020; Freitag et al., 2021a) and lead to a much more reliable way to assess MT quality (Kocmi et al., 2021). Nonetheless, all these metrics output a single point estimate, with the exception of UA-COMET (Glushkova et al., 2021), which returns a confidence interval along with a quality estimate. Our work builds upon UA-COMET by proposing improved uncertainty quantification.

Uncertainty quantification Epistemic (model) uncertainty represents the limitations of the model's knowledge (Der Kiureghian and Ditlevsen, 2009). Uncertainty quantification methods such as Gaussian processes (Williams and Rasmussen, 1996) can capture epistemic uncertainty (Postels et al., 2021; van Amersfoort et al., 2021). Beck et al. (2016) pioneered the use of Gaussian processes for quality estimation, yet these methods are hard to integrate into the powerful neural network architectures underlying state-of-the art MT evaluation systems. In contrast, ensemble-based methods for estimating model variance are more easily applicable – this includes MC dropout (Gal and Ghahramani, 2016) and Deep Ensembles (Lakshminarayanan et al., 2017). Recently, Raghu et al. (2019), Hu et al. (2021), and Jain et al. (2021) have shown that it is possible to predict the out-of-sample error by training a direct epistemic uncertainty predictor on the errors of the main model. To the best of our

[1]Our code will be made publicly available.

knowledge, direct uncertainty prediction have not been examined on MT evaluation (or other NLP tasks). Contrary to epistemic uncertainty, aleatoric (data) uncertainty corresponds to the irreducible amount of prediction error(s), which is due to the noise present in the observed data. Kendall and Gal (2017) propose the use of heteroscedastic variance in the loss function. Wang et al. (2019) propose a test-time augmentation-based aleatoric uncertainty. They compare and combine it with epistemic uncertainty, and show that it provides more representative uncertainty estimates than dropout-based ones alone. Our paper takes inspiration on these techniques to estimate aleatoric noise in MT evaluation.

Annotator disagreement Several approaches have been proposed to understand and model annotator bias (Cohn and Specia, 2013; Hovy and Yang, 2021) and to leverage annotator disagreement in NLP applications (Sheng et al., 2008; Plank et al., 2014, 2016; Jamison and Gurevych, 2015; Pavlick and Kwiatkowski, 2019). Recently, soft-label multi-task learning objectives for classification tasks have been proposed by Fornaciari et al. (2021). Our Kullback-Leibler divergence minimization objective may be regarded as an extension of this approach for regression tasks, replacing (softmax) categoricals by Gaussian distributions.

Uncertainty in NLP There are several works applying uncertainty quantification techniques to NLP, most commonly for (structured) classification tasks. Fomicheva et al. (2020) uses MC dropout to model MT confidence, and Malinin and Gales (2020) studies structured uncertainty estimation in autoregressive tasks, including MT and speech recognition. Ye et al. (2021) models uncertainty in performance prediction of NLP systems. Mielke et al. (2019) applies heteroscedastic models to assess language difficulty, whereas Friedl et al. (2021) estimates aleatoric uncertainty in scientific peer reviewing. While our paper focus on a regression task, some of our techniques might apply more broadly to these problems.

3 Uncertainty in MT Evaluation

3.1 MT evaluation

Throughout, we denote by s a sentence in a source language, by t a translation into a target language, and by $\mathcal{R}$ a set of reference translations. A segment-level **MT evaluation system** $\mathcal{M}_{\mathrm{Q}}$ (also called a "translation quality metric") is a system that takes as input a triple $\langle s, t, \mathcal{R}\rangle$ and outputs a quality score $\hat{q} \in \mathbb{R}$, reflecting how accurate t is as a translation of s. When $\mathcal{R} = \varnothing$, the metric $\mathcal{M}_{\mathrm{Q}}$ is called reference-less; otherwise it is reference-based.

Current state-of-the-art evaluation metrics, such as COMET (Rei et al., 2020a) or BLEURT (Sellam et al., 2020a), are trained with supervision on corpora annotated with human judgments $q^* \in \mathbb{R}$, such as direct assessments (DA; Graham et al. 2013) or scores from multi-dimensional quality metric annotations (MQM; Lommel et al. 2014). This supervision encourages their predicted quality scores to approximate the human perceived quality, $\hat{q} \approx q^*$, in a way that generalizes to unseen data.

3.2 Sources of uncertainty

While neural-based MT systems are more accurate than traditional lexical-based metrics such as BLEU, they are less transparent and may produce unreliable scores for out-of-domain inputs or when references are noisy (Rei et al., 2020b; Freitag et al., 2021b). Our goal is to mitigate this problem by quantifying the **uncertainty** associated with their predicted scores. This uncertainty can come from several sources:

- **Aleatoric (data) uncertainty** is primarily caused by noise in the data. Frequent sources of noise are inaccurate or inconsistent ground truth quality scores q^* (usually noticeable from low inter-annotator agreement scores) and noisy reference translations $\mathcal{R}$, which can mislead the MT evaluation system (Freitag et al., 2020).

- **Epistemic (model) uncertainty** reflects lack of knowledge from the model itself. This may be caused by limited training data, out-of-distribution examples (e.g., new languages, new domains, or diverse scoring schemes), or by complex, highly non-literal, translations which may trigger weak spots in the MT evaluation model.

Recently, Glushkova et al. (2021) proposed an **uncertainty-aware** evaluation metric (UA-COMET) by experimenting with two simple uncertainty quantification techniques, MC dropout (Gal and Ghahramani, 2016) and deep ensembles (Lakshminarayanan et al., 2017). Both techniques compute estimates based on **model variance** – they estimate uncertainty by running multiple versions of the system (either produced on-the-fly with stochastic dropout noise or by using separate models trained with different seeds), and then computing the mean $\hat{\mu}$ and variance $\hat{\sigma}^2$ of the predicted scores. When given a triple $\langle s, t, \mathcal{R} \rangle$ as input, instead of returning a point estimate $\hat{q}$, UA-COMET treats the quality score as a random variable Q, modeled as a Gaussian distribution $p_Q(q) = \mathcal{N}(q; \hat{\mu}, \hat{\sigma}^2)$. After a calibration step, the variance parameter of the Gaussian $\hat{\sigma}^2$ is used as the uncertainty estimate.

4 Improving Uncertainty-Aware MT Evaluation

A limitation of UA-COMET is that it relies on model variance techniques which often produce poor estimates of uncertainty and conflate aleatoric and epistemic uncertainty, making it hard to accurately represent uncertainty related to out-of-distribution samples (Jain et al., 2021; Zhang et al., 2021). We therefore examine alternate methods to learn aleatoric and epistemic uncertainty directly from the available data. We assume that for each of the training scenarios and learning objectives described in the following sections, we can learn to predict the uncertainty of quality estimates $\hat{q}$ either as the noise variance σ in the case of aleatoric uncertainty, or as the generalization error ϵ in the case of epistemic (and total) uncertainty.

4.1 Predicting aleatoric uncertainty

Rather than a property of the model, aleatoric uncertainty is a property of the data distribution and thus it can be learned as a function of the data (Kendall and Gal, 2017). It corresponds to uncertainty induced due to noise and inconsistencies. In the case of MT evaluation, we identify low quality references and inconsistent human annotations as the main sources of aleatoric uncertainty. The uncertainty associated with each data instance can vary: references have shown to be of different quality levels (Freitag et al., 2020), while the quality scores depend largely on the annotators and tend to have high disagreement (Toral, 2020).

Heteroscedasticity A common assumption in regression problems (of which MT evaluation is an example) is that the noise in the data has constant variance throughout the dataset – i.e., that the data is *homoscedastic*. The mean squared error loss, for example, corresponds to the maximum likelihood criterion under Gaussian noise with fixed variance. However, this is not a suitable assumption in several problems, including MT evaluation, where real data is often **heteroscedastic** – for example, complex sentences requiring specific background knowledge may be subject to larger annotation errors (higher disagreement among annotators) and higher chance for noisy references than simpler sentences. Therefore, the aleatoric uncertainty will likely be larger for those cases.

Heteroscedastic regression We model aleatoric uncertainty as observation noise by training a model to predict not only a quality score for each triple, but also a variance estimate $\hat{\sigma}^2$ for this score. Under our heteroscedastic assumption, we assume that the variance is specific to each data sample and can be learned as a function of the data. We follow Le et al. (2005) and Kendall and Gal (2017) and incorporate $\hat{\sigma}^2$ as part of the training objective, while learning the MT evaluation model parameters.

Formally, let $x := \langle s, t, \mathcal{R} \rangle$ denote an input triple, as described in §3. Our heteroscedastic uncertainty-aware MT evaluation system $\mathcal{M}_{\mathrm{Q}}^{\mathrm{HTS}}$ is a neural network that takes x as input and outputs a mean score $\hat{\mu}(x)$ and a variance score $\hat{\sigma}^2(x)$ – in practice, this is done by taking a COMET model and changing the output layer to output two scores ($\hat{\mu}(x)$ and $\log \hat{\sigma}^2(x)$) instead of one ($\hat{q}(x)$). This predicted mean and variance parametrize a Gaussian distribution $\hat{p}_Q(q|x;\theta) = \mathcal{N}(q; \hat{\mu}(x;\theta), \hat{\sigma}^2(x;\theta))$, where θ are the model parameters. Given a training set $\mathcal{D} = \{(x_1, q_1^*), \ldots, (x_N, q_N^*)\}$, the maximum likelihood training criterion amounts to maximize

$$\frac{1}{N} \sum_{i=1}^{N} \log \underbrace{\mathcal{N}(q_i^*; \hat{\mu}(x_i,\theta), \hat{\sigma}^2(x_i,\theta))}_{p_Q(q_i^*|x_i;\theta)} = \quad (1)$$

$$= -\frac{1}{N} \sum_{i=1}^{N} \mathcal{L}_{\mathrm{HTS}}(\hat{\mu}(x_i,\theta), \hat{\sigma}^2(x_i,\theta); q_i^*) + \mathrm{const.},$$

where $\mathcal{L}_{\mathrm{HTS}}$ denotes the **heteroscedastic loss**:

$$\mathcal{L}_{\mathrm{HTS}}(\hat{\mu}, \hat{\sigma}^2; q^*) = \frac{(q^* - \hat{\mu})^2}{2\hat{\sigma}^2} + \frac{1}{2} \log \hat{\sigma}^2. \quad (2)$$

We can see that, if $\hat{\sigma}^2$ was constant and not estimated, the heteroscedastic loss $\mathcal{L}_{\mathrm{HTS}}$ would revert to a standard squared loss; however, since this variance is predicted by the model and changes with the input, the model is trained to make a trade-off: the $\hat{\sigma}^2$ term in the denominator down-weights examples where the target q^* is assumed unreliable, decreasing the impact of highly noisy instances (a form of weighted least squares), while the $\log \hat{\sigma}^2$ term penalizes the model if it overestimates the variance. We show in §5.5 how this variance can be used to detect possibly noisy references.

KL divergence minimization While heteroscedastic uncertainty allows to estimate the observation noise, when we have multiple annotations for the same example we may have additional information on data uncertainty reflected in **annotator disagreement**. We assume that annotator disagreement in this case can be used as a proxy to data uncertainty.

Similarly to the estimation of heteroscedastic variance with the $\mathcal{L}_{\text{HTS}}$ objective, we assume that we can learn the variance $\hat{\sigma}(x;\theta)$ as an estimator of aleatoric uncertainty alongside the rest of the model, but now leveraging the supervision coming from the annotator disagreement – we denote this system by $\mathcal{M}_{\text{Q}}^{\text{KL}}$. We model the annotator scores as another Gaussian distribution $p_Q^*(q \mid x) = \mathcal{N}(q; \mu^*(x), \sigma^*(x))$, where $\mu^*(x)$ is the sample mean and $\sigma^*(x)$ the sample variance of the annotator scores for the example x, used as targets for our model predictions. We formalize this as a Kullback-Leibler (KL) divergence objective between the target distribution p_Q^* and the predicted distribution $\hat{p}_Q$, which has the following closed form for Gaussian distributions:

$$\begin{aligned}\mathcal{L}_{\text{KL}}(\hat{\mu}, \hat{\sigma}^2; \mu^*, {\sigma^*}^2) &= \text{KL}(p_Q^* \| \hat{p}_Q) \\ &= \frac{(\mu^* - \hat{\mu})^2 + {\sigma^*}^2}{2\hat{\sigma}^2} + \frac{1}{2}\log\frac{\hat{\sigma}^2}{{\sigma^*}^2} - \frac{1}{2}. \end{aligned} \tag{3}$$

Note that Eq. 3 is a generalization of Eq. 2: if we assume a fixed zero-limit variance ${\sigma^*}^2 \to 0$, we recover Eq. 2 up to a constant.

4.2 Predicting epistemic uncertainty

Epistemic (model) uncertainty can be observed mainly on out-of-sample and out-of-distribution instances, and manifests as the *reducible* generalization error of the model – in the presence of infinite training data and suitable model and learning algorithm, epistemic uncertainty could be reduced to zero (Postels et al., 2021; Jain et al., 2021). We outline two procedures to estimate epistemic and total uncertainty, one combining MC dropout with the heteroscedastic loss (Kendall and Gal, 2017), and another which estimates uncertainty directly as the generalization error (Jain et al., 2021).

Heteroscedastic MC dropout Given a way to estimate aleatoric uncertainty $\hat{\sigma}$, e.g., using Eqs. 2 or 3, we can combine it with an estimator of epistemic uncertainty to obtain the total uncertainty over a sample. Assuming we have access to an MT evaluation model that is able to predict both a quality score $\hat{q}$ and an aleatoric uncertainty estimate $\hat{\sigma}$ – such as the system $\mathcal{M}_{\text{Q}}^{\text{HTS}}$ described in §4.1 – we can use a stochastic strategy such as MC dropout or deep ensembles to obtain a set $\mathcal{Q} = \{\hat{q}_1, \dots, \hat{q}_M\}$ of quality estimates and $\Sigma = \{\hat{\sigma}_1^2, \dots, \hat{\sigma}_M^2\}$ of variance estimates. Assuming $\mathcal{Q}$ is a sample drawn from a Gaussian distribution, the sample variance can be used as an estimator of epistemic uncertainty, and the sample mean of Σ can be used as an estimator of aleatoric uncertainty (Kendall and Gal, 2017). We can then estimate the total uncertainty over the M samples as the sum of epistemic and aleatoric uncertainties:

$$\begin{aligned}\hat{\mathcal{U}}_{\text{total}} &= \text{Var}[\mathcal{Q}] + \mathbb{E}[\Sigma] \qquad (4) \\ &= \underbrace{\frac{1}{M}\sum_{j=1}^{M}\hat{q}_j^2 - \left(\frac{1}{M}\sum_{j=1}^{M}\hat{q}_j\right)^2}_{\text{epistemic}} + \underbrace{\frac{1}{M}\sum_{j=1}^{M}\hat{\sigma}_j^2}_{\text{aleatoric}}.\end{aligned}$$

For the experiments presented in §5 we use this strategy with MC dropout applied to a model trained with heteroscedastic regression.

Direct prediction of total uncertainty An alternative is to consider the total uncertainty $\hat{\mathcal{U}}_{\text{total}}$ as an approximation of the **generalization error** of the MT evaluation model $\mathcal{M}_{\text{Q}}$. In this case, assuming access to $\mathcal{M}_{\text{Q}}$'s predictions $\hat{q}$ and the ground truth quality scores q^* on a new (unseen) set of samples, we could learn to predict the total uncertainty **directly** as the error ϵ between the model predictions $\hat{q}$ and the true scores q^*, using the strategy recently proposed by Jain et al. (2021).

As opposed to the previously described uncertainty estimation approaches, direct uncertainty prediction (DUP) is a **two-step process**, as we need to first obtain the model $\mathcal{M}_{\text{Q}}$ that generates the predictions $\hat{q}$ that will allow us to estimate the target errors in a second stage. Hence, we need access to two distinct datasets on which two separate models have to be trained. We assume a dataset $\mathcal{D}_{\text{Q}}$ where $\mathcal{M}_{\text{Q}}$ is trained (we use the vanilla COMET system), and another, disjoint dataset $\mathcal{D}_{\text{E}}$ where we train a second system $\mathcal{M}_{\text{E}}$ to predict the uncertainty/error of $\mathcal{M}_{\text{Q}}$'s predictions. For this purpose, we use $\mathcal{M}_{\text{Q}}$ to annotate $\mathcal{D}_{\text{E}}$ with quality estimates $\hat{q}$, and then we calculate the ground truth error ϵ^* as the distance to the human quality scores q^* for each segment in $\mathcal{D}_{\text{E}}$, $\epsilon^* = |\hat{q} - q^*|$. We use ϵ^* as the target to train $\mathcal{M}_{\text{E}}$, given inputs $\langle s, t, \mathcal{R}, \hat{q}\rangle$. Letting $\hat{\epsilon}$ correspond to the uncertainty predicted by $\mathcal{M}_{\text{E}}$

on a given input, we consider three possible loss functions for $\mathcal{M}_{\mathrm{E}}$:

$$\mathcal{L}^{\mathrm{E}}_{\mathrm{ABS}}(\hat{\epsilon};\epsilon^*) = (\epsilon^* - \hat{\epsilon})^2 \quad (5)$$

$$\mathcal{L}^{\mathrm{E}}_{\mathrm{SQ}}(\hat{\epsilon};\epsilon^*) = ((\epsilon^*)^2 - \hat{\epsilon}^2)^2 \quad (6)$$

$$\mathcal{L}^{\mathrm{E}}_{\mathrm{HTS}}(\hat{\epsilon};\epsilon^*) = \frac{(\epsilon^*)^2}{2\hat{\epsilon}^2} + \frac{1}{2}\log(\hat{\epsilon})^2. \quad (7)$$

Losses $\mathcal{L}^{\mathrm{E}}_{\mathrm{ABS}}$ and $\mathcal{L}^{\mathrm{E}}_{\mathrm{SQ}}$ are variations of the mean squared error loss, using as argument either the absolute error $\hat{\epsilon}$ or the squared error $\hat{\epsilon}^2$. Instead, $\mathcal{L}^{\mathrm{E}}_{\mathrm{HTS}}$ is inspired by the heteroscedastic loss of Eq. 2, where the model is discouraged from predicting too high uncertainty values because of the term $\log(\hat{\epsilon})^2$, while it will still try to predict high $\hat{\epsilon}$ values for the samples where the MT quality score is not close to the human evaluation. Therefore, this choice is akin to a two-step approach to heteroscedastic regression: one step to train the "mean" predictor and another step for training the variance predictor given the mean predictions, where the two steps are performed on different partitions of the dataset, $\mathcal{D}_{\mathrm{Q}}$ and $\mathcal{D}_{\mathrm{E}}$.

5 Experiments

5.1 Experimental Setup

We follow Glushkova et al. (2021) and use COMET (v1.0) as the underlying architecture for our MT evaluation models, trained on the data from the WMT17-WMT19 metrics shared task (Freitag et al., 2021b). We consider two types of human judgments: direct assessments (DA) and multi-dimensional quality metric scores (MQM).

Experiments on DA scores We create a test partition with 20% of the WMT20 data (32,173 triplets).[2] All single-step models are trained on the data from the WMT17-WMT19 metrics shared task (WMT1719) and use the remaining 80% of WMT20 as a development set for calibration. For DUP models, WMT1719 is used to train the first step model $\mathcal{M}_{\mathrm{Q}}$ and the 80% split of WMT20 is used as follows: 70% to train DUP's second step model $\mathcal{M}_{\mathrm{E}}$ and 10% as development set. The data encompasses 16 language pairs (listed in Tables 4–5 in App. A), which we aggregate into two groups, EN-XX (out-of-English) and XX-EN (into-English). We report results for each group, as well as the balanced average across all language pairs (AVG).

Experiments on MQM scores We fine-tune all models on the entire WMT20 MQM dataset, which consists of MQM annotations for English-German (EN-DE) and Chinese-English (ZH-EN). For DUP we finetune the $\mathcal{M}_{\mathrm{E}}$ model on WMT20. For testing and calibration we use WMT21 metrics shared task dataset, which contains MQM annotations for the same language pairs, but also with an addition of English-Russian (EN-RU). We split the WMT21 MQM data into two halves, where 50% is used as a development set for calibrating all models, and 50% is used as the test set. We also provide the performance on the same WMT21 test set without any finetuning on MQM scores in the App. B.

Models As baselines, we use MC dropout (**MCD**) model with 100 dropout runs, and a deep ensemble (**DE**) of 5 independent COMET models. We experiment with the following models: an heteroscedastic COMET model $\mathcal{M}^{\mathrm{HTS}}_{\mathrm{Q}}$ trained with the loss in Eq. 2 (**HTS**), its combination with MC dropout as described in Eq. 4 (**HTS+MCD**), and the direct uncertainty prediction model described in §4.2 (**DUP**) using the three losses in Eqs. 5–7. For the DUP models, we use vanilla COMET as $\mathcal{M}_{\mathrm{Q}}$ and a system with the same architecture for $\mathcal{M}_{\mathrm{E}}$ which receives as an additional feature the predicted quality score $\hat{q}$ from $\mathcal{M}_{\mathrm{Q}}$. This extra feature is added by inserting a bottleneck layer between two feed-forward layers in the original vanilla COMET architecture (see App. C). Finally, for the experiment with MQM scores, where multiple annotators for the same examples are available, we also experiment with the model $\mathcal{M}^{\mathrm{KL}}_{\mathrm{Q}}$ using the objective in Eq. 3 (**KL**).[3]

Evaluation For both types of human judgements (DA and MQM), in all the experiments, we report the same performance indicators as Glushkova et al. (2021): the predictive Pearson score $r(\hat{\mu}, q^*)$ (PPS), the uncertainty Pearson score $r(|q^* - \hat{\mu}|, \hat{\sigma})$ (UPS), the negative log-likelihood $-\log\mathcal{N}(q^*; \hat{\mu}, \hat{\sigma}^2)$ (NLL), the expected calibration error (ECE), and the sharpness (Sha.), i.e., the average predicted variance in the test set. These indicators are described in detail in App. D; they

[2] We ensure that triplets with the same source sentence or from the same document do not appear in the other sets so that these sets are disjoint. All sets are balanced with respect to the percentage of source segments available from each language pair. The splits will be made publicly available.

[3] Unlike the other models, the KL model is trained directly on the WMT20 MQM dataset (instead of being just fine-tuned there), since the WMT data with direct assessments does not include information on annotator disagreement that is used as target for the KL model training.

		PPS ↑	UPS ↑	NLL ↓	ECE ↓	Sha. ↓
EN-XX	DUP $\mathcal{L}^{\mathrm{E}}_{\mathrm{ABS}}$	0.633	0.134	1.019	0.013	0.295
	DUP $\mathcal{L}^{\mathrm{E}}_{\mathrm{SQ}}$	0.633	0.140	1.022	0.012	0.315
	DUP $\mathcal{L}^{\mathrm{E}}_{\mathrm{HTS}}$	0.633	0.146	1.021	0.014	0.293
XX-EN	DUP $\mathcal{L}^{\mathrm{E}}_{\mathrm{ABS}}$	0.287	0.081	1.471	0.017	0.527
	DUP $\mathcal{L}^{\mathrm{E}}_{\mathrm{SQ}}$	0.287	0.084	1.470	0.017	0.534
	DUP $\mathcal{L}^{\mathrm{E}}_{\mathrm{HTS}}$	0.287	0.086	1.473	0.017	0.524
AVG	DUP $\mathcal{L}^{\mathrm{E}}_{\mathrm{ABS}}$	0.446	0.104	1.265	0.015	0.414
	DUP $\mathcal{L}^{\mathrm{E}}_{\mathrm{SQ}}$	0.446	0.108	1.262	0.014	0.427
	DUP $\mathcal{L}^{\mathrm{E}}_{\mathrm{HTS}}$	0.446	0.112	1.266	0.015	0.411

Table 1: Comparison of different losses for the DUP method in segment-level DA prediction.

		PPS ↑	UPS ↑	NLL ↓	ECE ↓	Sha. ↓
EN-XX	MCD	0.601	0.128	0.616	0.032	0.673
	DE	0.631	0.134	$\underline{0.522}$	0.086	0.461
	HTS	0.617	0.172	0.911	0.026	0.377
	HTS+MCD	0.609	$\underline{0.323}$	0.994	0.017	0.516
	DUP	$\underline{0.633}$	0.145	1.021	$\underline{0.013}$	$\underline{0.293}$
XX-EN	MCD	0.286	0.019	1.033	0.073	1.432
	DE	$\underline{0.296}$	0.044	$\underline{0.943}$	0.086	1.131
	HTS	0.278	0.079	1.441	$\underline{0.011}$	0.566
	HTS+MCD	0.276	$\underline{0.176}$	1.323	$\underline{0.011}$	0.600
	DUP	0.287	0.086	1.473	0.017	$\underline{0.524}$
AVG	MCD	0.435	0.071	0.816	0.053	1.083
	DE	$\underline{0.455}$	0.090	$\underline{0.728}$	0.086	0.813
	HTS	0.435	0.119	1.189	0.020	0.466
	HTS+MCD	0.429	$\underline{0.254}$	1.167	$\underline{0.013}$	0.528
	DUP	0.446	0.112	1.266	0.015	$\underline{0.411}$

Table 2: Results for segment-level DA predictions. Underlined numbers indicate the best result for each evaluation metric in each language pair.

assess both quality prediction accuracy (PPS), uncertainty-related accuracy (UPS, ECE and Sha.), and the two combined in a single score (NLL).

5.2 Loss function for DUP

We first compare the performance of the three aforementioned losses for **DUP** (see Eqs. 5–7 in §4.2) on the segment-level DA data. According to the results in Table 1, all three losses perform similarly, with a slight advantage to $\mathcal{L}^{\mathrm{E}}_{\mathrm{HTS}}$. We thus run the rest of the experiments using this loss as a representative of **DUP**.

5.3 Comparison of uncertainty methods

The results of the DA and MQM experiments are shown in Tables 2–3. As expected, the PPS values (which do not measure uncertainty, but accuracy of the quality predictions) are similar for all methods, since they are based either on a vanilla COMET model, or on an ensemble of COMET models, with an advantage for the **DE** method which benefits from the ensemble effect. While **HTS** and **KL** have modified objectives that learn the mean and the variance simultaneously, they do not seem to improve the quality predictions. We focus our analysis in the uncertainty prediction, assessed by the other four indicators (UPS, NLL, ECE, and Sha.)

For the DA experiments, we observe that our two proposed methods, **HTS** and **DUP**, are consistently better than the baseline estimates (**MCD** and **DE**) for all uncertainty metrics (UPS, ECE, and Sha.) except NLL. The significant drop in NLL might be explained by the fact that **DUP** tends to underestimate the variance, and this is severely penalized by NLL (see App. D). Applying MC dropout to $\mathcal{M}^{\mathrm{HTS}}_{\mathrm{Q}}$ (**HTS+MCD**) seems to improve UPS and ECE, compared to $\mathcal{M}^{\mathrm{HTS}}_{\mathrm{Q}}$ (**HTS**) alone, but it produces less sharp uncertainty estimates and negatively impacts the predictive accuracy of the model. **DUP** on the other hand seems to outperform other methods and gets more informative and "tight" uncertainty intervals. Additionally, as we can see in Figure 1, the sharpness increases for out-of-domain data in the case of **DUP** and captures nicely the increased epistemic uncertainty in such cases. In contrast, we can see that variance based epistemic uncertainty predictors cannot accurately represent the domain shift, while aleatoric uncertainty (**HTS**) remains the same. We provide a more extended analysis of this aspect in the App. E.

The findings on DA data are further supported by the MQM results, and we can see that the models achieve good performance for the EN-RU language pair, which is not available in the WMT20 MQM data used for fine-tuning. We also see that the **KL** model, despite having access to significantly less training data (see §5.1), achieves results that are close to **DUP**, specially for the pairs EN-DE and ZH-EN where it was trained.

5.4 Computational cost

We now turn to the computational cost associated with the different uncertainty quantification methods, both in terms of training and inference runtime. In Figure 3, we present the inference and training times for each of the discussed models. The large inference times for **MCD** and **HTS+MCD** stem from the need to run 100 runs (the optimal number according to Glushkova et al. (2021)); for DE, 5 models are ensembled, increasing training and inference costs 5-fold (for training details see Tab. 7 in App. C). In contrast, **HTS**, **KL**, and **DUP** have much lighter costs (with higher costs for **DUP** due

		PPS ↑	UPS ↑	NLL ↓	ECE ↓	Sha. ↓
EN-DE	MCD	0.358	0.185	$\underline{0.631}$	0.085	0.952
	DE	$\underline{0.402}$	0.116	0.671	$\underline{0.019}$	1.034
	HTS	0.342	$\underline{0.303}$	2.001	0.029	0.200
	HTS+MCD	0.334	0.301	1.564	0.027	0.296
	KL	0.358	0.239	2.500	0.049	$\underline{0.188}$
	DUP	0.374	0.255	2.373	0.045	0.20
ZH-EN	MCD	0.618	0.237	0.944	0.101	1.032
	DE	$\underline{0.628}$	0.239	$\underline{0.869}$	0.025	0.894
	HTS	0.599	0.340	1.445	0.038	$\underline{0.487}$
	HTS+MCD	0.603	$\underline{0.357}$	1.314	0.037	0.544
	KL	0.587	0.316	1.564	0.036	0.990
	DUP	0.616	0.346	1.290	$\underline{0.012}$	0.492
EN-RU	MCD	0.364	0.213	0.69	0.09	1.11
	DE	$\underline{0.376}$	0.281	$\underline{0.363}$	0.06	1.087
	HTS	0.336	0.302	2.564	0.075	$\underline{0.15}$
	HTS+MCD	0.357	$\underline{0.349}$	1.622	0.051	0.355
	KL	0.353	0.275	4.046	$\underline{0.044}$	$\underline{0.15}$
	DUP	0.371	0.331	2.477	0.046	0.206
AVG	MCD	0.463	0.215	0.775	0.093	1.038
	DE	$\underline{0.482}$	0.222	$\underline{0.643}$	0.036	0.997
	HTS	0.441	0.317	1.976	0.048	$\underline{0.296}$
	HTS+MCD	0.448	$\underline{0.34}$	1.485	0.039	0.414
	KL	0.447	0.282	2.664	0.042	0.492
	DUP	0.469	0.317	1.981	$\underline{0.032}$	0.317

Table 3: Results for segment-level MQM predictions. Underlined numbers indicate the best result for each evaluation metric in each language pair.

to the need to train/run a second system).

5.5 Identification of noisy references

As mentioned in §3.2, low quality references are a primary source of aleatoric uncertainty. Thus, we expect the uncertainty predictors that model aleatoric uncertainty (**HTS** and **KL**) to be more sensitive to erroneous references compared to the other uncertainty predictors. To verify this hypothesis and investigate the potential of aleatoric uncertainty predictors to detect noisy references, we conduct an experiment on the WMT21 MQM EN-DE dataset, which includes 4 references, each annotated with MQM scores by a human annotator (Freitag et al., 2021b). We can thus use these MQM scores as indicators of how good references are. For each $\langle s, t \rangle$ pair in the test split, we select the best reference r_{good} and the worst reference r_{bad} based on the respective MQM scores. We retain only the $\langle s, t, \{r_{\text{good}}, r_{\text{bad}}\} \rangle$ for which $|\text{MQM}(r_{\text{good}}) - \text{MQM}(r_{\text{bad}})| > 10$,, so that there is a considerable quality difference between the references.[4] We then apply the uncertainty predictors on the selected triples $\langle s, t, r_{\text{good}} \rangle$ and $\langle s, t, r_{\text{bad}} \rangle$ and obtain the predicted uncertainties, as shown in Figure 2. For each $\langle s, t \rangle$ pair, we check which reference leads to the lowest predicted uncertainty and compute how often that reference coincides with r_{good}. In Figure 4, we can see that both the **HTS** and the **KL** predictors are much more successful in choosing the correct reference compared to **MCD** (**HTS** in particular is correct $> 82\%$ of the time versus 38% for **MCD**). This confirms the hypothesis that **HTS** and **KL** are more effective at capturing aleatoric uncertainty.

[4] An MQM penalty of 10 points corresponds to at least 2 major errors (Freitag et al., 2021a).

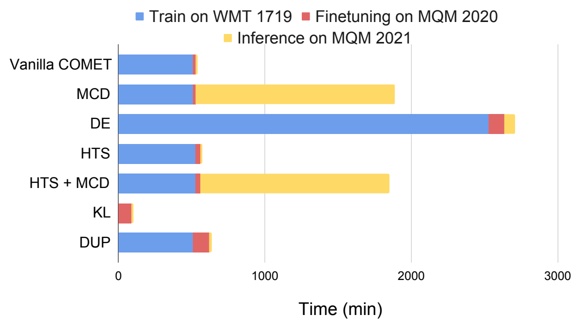


Figure 3: Combined training, fine-tuning and inference times for the experiments reported in Table 3. All experiments were performed on a server with 4 Quadro RTX 6000 (24GB), 12 Intel Xeon Silver 4214@2.20GHz CPUs, and 256 Gb of RAM; time calculated for training/inference on a single GPU.

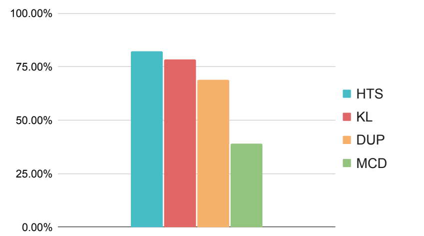


Figure 4: Percentage of correctly recognized higher reference quality (r_{good} versus r_{bad}) by different uncertainty predictors on the EN-DE dataset.

6 Conclusions

We explored the potential of different uncertainty predictors to capture different sources of uncertainty in MT evaluation. We demonstrated that methods modeling heteroscedasticity are useful for detecting noisy references as a source of aleatoric uncertainty, and that the direct epistemic prediction method reflects well the increased epistemic uncertainty under a domain shift. Our proposed predictors, besides providing more informative uncertainty estimates than MC dropout and deep ensemble methods, are also considerably cheaper in terms of computational costs.

References

Daniel Beck, Lucia Specia, and Trevor Cohn. 2016. Exploring prediction uncertainty in machine translation quality estimation. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 208–218, Berlin, Germany. Association for Computational Linguistics.

Trevor Cohn and Lucia Specia. 2013. Modelling annotator bias with multi-task gaussian processes: An application to machine translation quality estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 32–42.

Armen Der Kiureghian and Ove Ditlevsen. 2009. Aleatory or epistemic? does it matter? *Structural safety*, 31(2):105–112.

Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. 2020. Unsupervised quality estimation for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:539–555.

Tommaso Fornaciari, Alexandra Uma, Silviu Paun, Barbara Plank, Dirk Hovy, and Massimo Poesio. 2021. Beyond black & white: Leveraging annotator disagreement via soft-label multi-task learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2591–2597, Online. Association for Computational Linguistics.

Markus Freitag, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021a. Experts, Errors, and Context: A Large-Scale Study of Human Evaluation for Machine Translation. *Transactions of the Association for Computational Linguistics*, 9:1460–1474.

Markus Freitag, David Grangier, and Isaac Caswell. 2020. BLEU might be guilty but references are not innocent. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 61–71, Online. Association for Computational Linguistics.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, George Foster, Alon Lavie, and Ondřej Bojar. 2021b. Results of the WMT21 metrics shared task: Evaluating metrics with expert-based human evaluations on TED and news domain. In *Proceedings of the Sixth Conference on Machine Translation*, pages 733–774, Online. Association for Computational Linguistics.

Korbinian Friedl, Georgios Rizos, Lukas Stappen, Madina Hasan, Lucia Specia, Thomas Hain, and Björn Schuller. 2021. Uncertainty aware review hallucination for science article classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 5004–5009, Online. Association for Computational Linguistics.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059. PMLR.

Taisiya Glushkova, Chrysoula Zerva, Ricardo Rei, and André F. T. Martins. 2021. Uncertainty-aware machine translation evaluation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3920–3938, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yvette Graham, Timothy Baldwin, Alistair Moffat, and Justin Zobel. 2013. Continuous measurement scales in human evaluation of machine translation. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 33–41, Sofia, Bulgaria. Association for Computational Linguistics.

Dirk Hovy and Diyi Yang. 2021. The importance of modeling social factors of language: Theory and practice. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 588–602.

Shi Hu, Nicola Pezzotti, and Max Welling. 2021. Learning to predict error for mri reconstruction. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 604–613. Springer.

Moksh Jain, Salem Lahlou, Hadi Nekoei, Victor Butoi, Paul Bertin, Jarrid Rector-Brooks, Maksym Korablyov, and Yoshua Bengio. 2021. Deup: Direct epistemic uncertainty prediction. *arXiv preprint arXiv:2102.08501*.

Emily Jamison and Iryna Gurevych. 2015. Noise or additional information? leveraging crowdsource annotation item agreement for natural language tasks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 291–297.

Alex Kendall and Yarin Gal. 2017. What uncertainties do we need in bayesian deep learning for computer vision? In *NIPS*.

Fabio Kepler, Jonay Trénous, Marcos Treviso, Miguel Vera, and André F. T. Martins. 2019. OpenKiwi: An open source framework for quality estimation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 117–122, Florence, Italy. Association for Computational Linguistics.

Tom Kocmi, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494, Online. Association for Computational Linguistics.

Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. 2018. Accurate uncertainties for deep learning using calibrated regression. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2796–2804. PMLR.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Alon Lavie and Michael Denkowski. 2009. The Meteor metric for automatic evaluation of Machine Translation. *Machine Translation*, 23:105–115.

Quoc V. Le, Alex J. Smola, and Stéphane Canu. 2005. Heteroscedastic gaussian process regression. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, page 489–496, New York, NY, USA. Association for Computing Machinery.

Chi-kiu Lo. 2019. YiSi - a unified semantic MT quality evaluation and estimation metric for languages with different levels of available resources. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 507–513, Florence, Italy. Association for Computational Linguistics.

Arle Lommel, Aljoscha Burchardt, and Hans Uszkoreit. 2014. Multidimensional quality metrics (MQM): A framework for declaring and describing translation quality metrics. *Tradumàtica: tecnologies de la traducció*, 0:455–463.

Andrey Malinin and Mark Gales. 2020. Uncertainty estimation in autoregressive structured prediction. *arXiv preprint arXiv:2002.07650*.

Nitika Mathur, Johnny Wei, Markus Freitag, Qingsong Ma, and Ondřej Bojar. 2020. Results of the WMT20 metrics shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 688–725, Online. Association for Computational Linguistics.

Sabrina J. Mielke, Ryan Cotterell, Kyle Gorman, Brian Roark, and Jason Eisner. 2019. What kind of language is hard to language-model? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4975–4989, Florence, Italy. Association for Computational Linguistics.

Mahdi Pakdaman Naeini, Gregory F. Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, page 2901–2907. AAAI Press.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Ellie Pavlick and Tom Kwiatkowski. 2019. Inherent disagreements in human textual inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694.

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014. Learning part-of-speech taggers with inter-annotator agreement loss. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 742–751.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 412–418, Berlin, Germany. Association for Computational Linguistics.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Janis Postels, Mattia Segu, Tao Sun, Luc Van Gool, Fisher Yu, and Federico Tombari. 2021. On the practicality of deterministic epistemic uncertainty. *arXiv preprint arXiv:2107.00649*.

Maithra Raghu, Katy Blumer, Rory Sayres, Ziad Obermeyer, Bobby Kleinberg, Sendhil Mullainathan, and Jon Kleinberg. 2019. Direct uncertainty prediction for medical second opinions. In *International Conference on Machine Learning*, pages 5281–5290. PMLR.

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2020. TransQuest: Translation quality estimation with cross-lingual transformers. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5070–5081, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020a. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020b. Unbabel's participation in the WMT20 metrics shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 911–920, Online. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020a. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Thibault Sellam, Amy Pu, Hyung Won Chung, Sebastian Gehrmann, Qijun Tan, Markus Freitag, Dipanjan Das, and Ankur Parikh. 2020b. Learning to evaluate translation beyond English: BLEURT submissions to the WMT metrics 2020 shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 921–927, Online. Association for Computational Linguistics.

Victor S Sheng, Foster Provost, and Panagiotis G Ipeirotis. 2008. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 614–622.

Brian Thompson and Matt Post. 2020. Automatic machine translation evaluation in many languages via zero-shot paraphrasing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 90–121, Online. Association for Computational Linguistics.

Antonio Toral. 2020. Reassessing claims of human parity and super-human performance in machine translation at WMT 2019. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 185–194, Lisboa, Portugal. European Association for Machine Translation.

Joost van Amersfoort, Lewis Smith, Andrew Jesson, Oscar Key, and Yarin Gal. 2021. Improving deterministic uncertainty estimation in deep learning for classification and regression. *CoRR*, abs/2102.11409.

Guotai Wang, Wenqi Li, Michael Aertsen, Jan Deprest, Sébastien Ourselin, and Tom Vercauteren. 2019. Aleatoric uncertainty estimation with test-time augmentation for medical image segmentation with convolutional neural networks. *Neurocomputing*, 338:34–45.

CKI Williams and CE Rasmussen. 1996. Gaussian processes for regression. In *Ninth Annual Conference on Neural Information Processing Systems (NIPS 1995)*, pages 514–520. MIT Press.

Zihuiwen Ye, Pengfei Liu, Jinlan Fu, and Graham Neubig. 2021. Towards more fine-grained and reliable NLP performance prediction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3703–3714, Online. Association for Computational Linguistics.

Jing Zhang, Yuchao Dai, Mochu Xiang, Deng-Ping Fan, Peyman Moghadam, Mingyi He, Christian Walder, Kaihao Zhang, Mehrtash Harandi, and Nick Barnes. 2021. Dense uncertainty estimation. *arXiv preprint arXiv:2110.06427*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A DA experiments

Results per language pair are presented in Tables 4 and 5.

		PPS ↑	UPS ↑	NLL ↓	ECE ↓	Sha. ↓
EN-CS	MCD	0.702	0.108	0.587	0.055	0.432
	DE	0.753	0.119	0.561	0.089	0.446
	HTS	0.71	0.195	0.807	0.004	0.353
	HTS+MCD	0.676	0.335	0.88	0.008	0.388
	DUP	0.718	0.132	0.987	0.007	0.559
EN-DE	MCD	0.432	0.21	0.562	0.019	0.592
	DE	0.479	0.198	0.365	0.08	0.348
	HTS	0.632	0.253	0.925	0.021	0.326
	HTS+MCD	0.612	0.386	0.924	0.004	0.369
	DUP	0.655	0.166	1.045	0.031	0.272
EN-JA	MCD	0.697	0.192	0.596	0.013	0.581
	DE	0.714	0.128	0.425	0.09	0.348
	HTS	0.672	0.197	0.8	0.057	0.406
	HTS+MCD	0.669	0.249	1.048	0.03	0.52
	DUP	0.698	0.153	0.85	0.009	0.158
EN-PL	MCD	0.624	0.097	0.834	0.038	1.078
	DE	0.66	0.092	0.783	0.089	0.704
	HTS	0.628	0.073	1.281	0.008	0.349
	HTS+MCD	0.631	0.189	0.969	0.004	0.397
	DUP	0.62	0.046	1.551	0.008	0.316
EN-RU	MCD	0.519	0.124	0.587	0.036	0.562
	DE	0.544	0.142	0.541	0.086	0.444
	HTS	0.493	0.163	0.877	0.005	0.385
	HTS+MCD	0.497	0.37	1.05	0.017	0.637
	DUP	0.512	0.229	0.863	0.013	0.287
EN-TA	MCD	0.639	0.031	0.832	0.041	1.18
	DE	0.656	0.078	0.772	0.088	0.704
	HTS	0.623	0.185	1.112	0.005	0.485
	HTS+MCD	0.617	0.231	1.223	0.038	0.921
	DUP	0.641	0.172	1.164	0.016	0.348
EN-ZH	MCD	0.592	0.131	0.313	0.024	0.282
	DE	0.612	0.178	0.207	0.082	0.235
	HTS	0.566	0.139	0.58	0.083	0.337
	HTS+MCD	0.562	0.504	0.865	0.02	0.38
	DUP	0.586	0.122	0.688	0.011	0.11

Table 4: Results for segment-level DA prediction for En-Xx LPs. Underlined numbers indicate the best result for each evaluation metric in each language pair.

B MQM experiments

Results without fine-tuning on the MQM data are presented in Table 6. For these experiments we use the models trained on the WMT DA data (performance for these models is also reported in Table 2). We can see that without further finetuning on MQM scores all models with the exception of the ones based on variance (**MCD** and **DE**) have a significant drop in performance.

C Model implementation and parameters

Table 7 shows the hyperparameters used to train the following uncertainty prediction models: **MCD**, **DE**, **HTS**, **KL** and **DUP**. For deep ensembles we trained 4 models with different seeds and as a fifth model we used the *wmt-comet-da* available at `https://github.com/Unbabel/COMET` (in the table we refer to it as **Vanilla COMET**).

D Performance indicators

We briefly describe below each of the metrics reported for the experiments of this paper, provide the formulas for each one and the motivation for using them. For all described metrics we assume access to a test set $\mathcal{D} = \{\langle s_j, t_j, \mathcal{R}_j, q_j^* \rangle\}_{j=1}^{|\mathcal{D}|}$, consisting of samples paired with their ground truth quality scores.

Calibration Error To estimate how well-calibrated the methods are we compute expected calibration error (ECE; Naeini et al. 2015; Kuleshov et al. 2018), which is defined as:

$$\mathrm{ECE} = \frac{1}{M} \sum_{b=1}^{M} |\mathrm{acc}(\gamma_b) - \gamma_b|, \qquad (8)$$

where each b is a bin representing a confidence level γ_b, and $\mathrm{acc}(\gamma_b)$ is the fraction of times the ground truth q^* falls inside the confidence interval $I(\gamma_b)$:

$$\mathrm{acc}(\gamma_b) = \frac{1}{|\mathcal{D}|} \sum_{\langle s,t,\mathcal{R},q^* \rangle \in \mathcal{D}} \mathbb{1}(q^* \in I(\gamma_b)). \quad (9)$$

We use this metric with $M = 100$, similarly to previous works.

Negative log-likelihood The negative log-likelihood (NLL) captures both accuracy- and uncertainty-related performance, since it essentially considers the log-likelihood of the true quality score q^* based on the distribution estimated by the predicted variance (uncertainty). Thus it penalizes predictions that are accurate but have too high uncertainty (since they will become flat distributions with low probability everywhere), and even more severely incorrect predictions with high confidence, but is more lenient with predictions that are inaccurate but have high uncertainty.

$$\mathrm{NLL} = -\frac{1}{|\mathcal{D}|} \sum_{\langle s,t,\mathcal{R},q^* \rangle \in \mathcal{D}} \log \hat{p}(q^* \mid \langle s, t, \mathcal{R} \rangle). \qquad (10)$$

Note that it is possible to calculate the optimal fixed variance that minimizes NLL by:

$$\sigma^2_{\text{fixed}} = \frac{1}{|\mathcal{D}|} \sum_{j=1}^{|\mathcal{D}|} (q_j^* - \hat{\mu}_j)^2. \qquad (11)$$

Sharpness To ensure informative uncertainty estimation, confidence intervals should not only be calibrated, but also sharp. We measure sharpness using the predicted variance $\hat{\sigma}^2$, as defined in Kuleshov et al. (2018):

$$\mathrm{sha}(\hat{p}_Q) = \frac{1}{|\mathcal{D}|} \sum_{\langle s,t,\mathcal{R}\rangle \in \mathcal{D}} \hat{\sigma}^2. \tag{12}$$

Pearson correlations The **predictive Pearson score** (PPS), evaluates the predictive accuracy of the system – it is the Pearson correlation $r(q^*, \hat{q})$ between the ground truth quality scores q^* and the system predictions $\hat{q}$ in the dataset $\mathcal{D}$. The **uncertainty Pearson score** (UPS) $r(|q^* - \hat{q}|, \hat{\sigma})$, measures the alignment between the prediction errors $|q^* - \hat{q}|$ and the uncertainty estimates $\hat{\sigma}$.

E Uncertainty on OOD examples

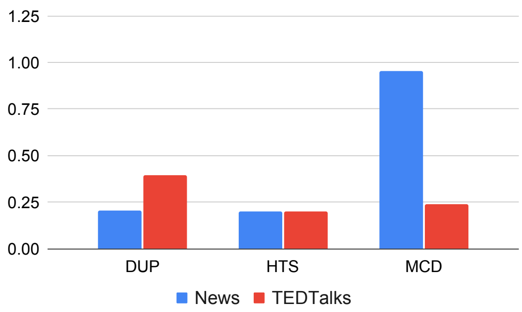
We provide the comparison of the sharpness value, representing the quantified uncertainty for in-domain (ID) data (WMT21 news data with MQM annotations) and out-of-domain (OOD) data (WMT21 TEDTalks data with MQM annotations) in Figure 5. Sharpness as explained in App. D, is an indicator of the overall estimated confidence of a model over a given dataset. Thus we want to examine whether the estimated confidence intervals for the OOD data are representative of the expected increase in epistemic uncertainty.

Looking at the sharpness variation per language pair, we can see that for EN-DE and EN-RU, where the aleatoric uncertainty is relatively low as indicated by the low HTS values, the sharpness increases significantly for the DUP model. This behaviour however does not hold for cases where aleatoric uncertainty is higher (ZH-EN). We speculate that this could be attributed to the fact that DUP is trained to capture total uncertainty, instead of only epistemic, and thus it is sensitive to increased noise in the data. Further experiments would be needed to verify this hypothesis.

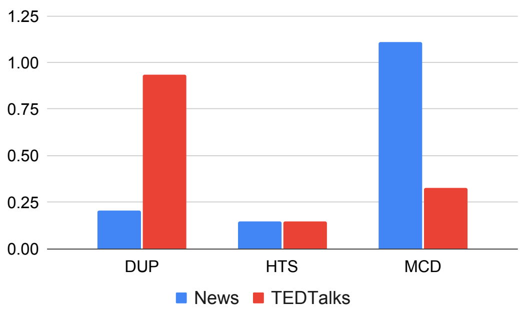
Across language pairs, the values for HTS remain the same for ID and OOD, while for MCD we have the opposite effect than what was expected: sharpness drops significantly for OOD data in all language pairs. This further supports our claim that uncertainty predictors relying on model variance are not optimal to represent epistemic uncertainty.

		PPS ↑	UPS ↑	NLL ↓	ECE ↓	Sha. ↓
CS-EN	MCD	0.194	0.035	0.912	0.052	1.463
	DE	0.208	0.052	0.903	0.086	1.064
	HTS	0.197	0.041	1.378	0.012	0.503
	HTS+MCD	0.195	0.229	1.276	0.006	0.502
	DUP	0.198	0.057	1.446	0.016	0.539
DE-EN	MCD	0.203	0.005	0.825	0.065	1.075
	DE	0.215	-0.011	0.734	0.081	0.855
	HTS	0.189	0.018	1.416	0.01	0.306
	HTS+MCD	0.163	0.148	1.107	0.014	0.463
	DUP	0.211	0.019	1.336	0.013	0.501
JA-EN	MCD	0.339	0.058	1.579	0.065	1.702
	DE	0.348	0.107	0.966	0.088	1.059
	HTS	0.338	0.189	1.306	0.009	0.731
	HTS+MCD	0.339	0.215	1.322	0.007	0.611
	DUP	0.333	0.157	1.365	0.019	0.502
KM-EN	MCD	0.46	-0.084	1.06	0.09	1.104
	DE	0.466	-0.014	1.029	0.094	1.061
	HTS	0.447	0.151	1.245	0.008	0.742
	HTS+MCD	0.452	0.143	1.263	0.015	0.836
	DUP	0.453	0.144	1.24	0.011	0.608
PL-EN	MCD	0.275	0.011	0.98	0.067	1.659
	DE	0.282	0.003	0.985	0.081	1.323
	HTS	0.269	0.03	1.598	0.01	0.562
	HTS+MCD	0.268	0.139	1.424	0.008	0.502
	DUP	0.277	0.074	1.641	0.01	0.591
PS-EN	MCD	0.321	0.048	1.093	0.094	1.24
	DE	0.327	0.034	1.085	0.096	1.201
	HTS	0.291	0.034	1.331	0.006	0.754
	HTS+MCD	0.297	0.11	1.315	0.013	0.849
	DUP	0.322	0.054	1.298	0.012	0.658
RU-EN	MCD	0.214	0.012	0.926	0.061	1.79
	DE	0.233	0.05	0.889	0.079	1.226
	HTS	0.219	0.056	1.767	0.021	0.418
	HTS+MCD	0.209	0.161	1.520	0.013	0.493
	DUP	0.223	0.039	1.839	0.029	0.38
TA-EN	MCD	0.276	0.07	0.966	0.085	1.25
	DE	0.282	0.104	0.98	0.084	1.219
	HTS	0.277	0.134	1.471	0.011	0.511
	HTS+MCD	0.284	0.25	1.300	0.016	0.642
	DUP	0.27	0.132	1.56	0.027	0.422
ZH-EN	MCD	0.29	0.014	0.952	0.073	1.598
	DE	0.303	0.069	0.916	0.085	1.172
	HTS	0.282	0.067	1.454	0.011	0.572
	HTS+MCD	0.278	0.186	1.377	0.006	0.504
	DUP	0.293	0.093	1.531	0.019	0.517

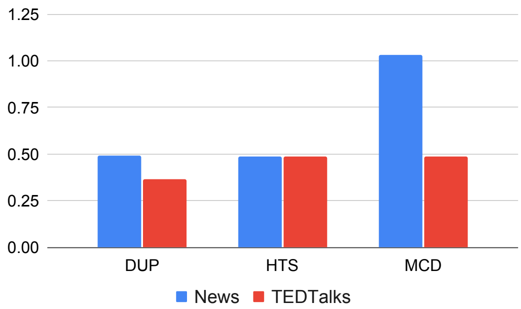
Table 5: Results for segment-level DA prediction for Xx-En LPs. Underlined numbers indicate the best result for each evaluation metric in each language pair.



(a) EN-DE



(b) EN-RU



(c) ZH-EN

Figure 5: Sharpness for in-domain (blue) News WMT21 MQM data and out-of-domain (red) TEDTalks WMT21 MQM data. We show changes in sharpness values on each language pair separately, for the **DUP**, **HTS** and **MCD** models finetuned on News WMT20 MQM data.

		PPS ↑	UPS ↑	NLL ↓	ECE ↓	Sha. ↓
EN-DE	MCD	0.295	0.134	0.577	0.069	1.019
	DE	0.332	0.104	0.644	0.021	1.03
	HTS	0.326	0.094	0.039	2.567	0.274
	HTS + MCD	0.291	0.126	1.502	0.021	0.356
	DUP	0.302	0.038	2.248	0.054	0.241
ZH-EN	MCD	0.441	0.115	0.956	0.081	1.321
	DE	0.457	0.14	0.911	0.025	1.143
	HTS	0.436	0.082	0.013	1.615	0.595
	HTS + MCD	0.433	-0.006	1.42	0.013	0.637
	DUP	0.434	0.17	1.814	0.05	0.469
EN-RU	MCD	0.306	0.14	0.563	0.069	1.242
	DE	0.318	0.117	0.684	0.078	1.332
	HTS	0.337	0.134	0.021	2.035	0.306
	HTS + MCD	0.333	-0.042	1.492	0.016	0.459
	DUP	0.290	0.139	2.238	0.045	0.35
AVG	MCD	0.356	0.129	0.722	0.074	1.215
	DE	0.377	0.123	0.763	0.042	1.179
	HTS	0.289	0.079	0.012	1.34	0.341
	HTS + MCD	0.286	-0.017	1.076	0.011	0.41
	DUP	0.272	0.115	1.489	0.035	0.306

Table 6: Results for segment-level MQM prediction. Underlined numbers indicate the best result for each evaluation metric in each language pair.

Hyperparameter	MCD/DE/Vanilla COMET	HTS/KL	DUP
Encoder Model	XLM-R (large)	XLM-R (large)	XLM-R (large)
Optimizer	Adam	Adam	Adam
No. frozen epochs	0.3	0.3	0.3
Learning rate	3e-05	3e-05	3e-05
Encoder Learning Rate	1e-05	1e-05	1e-05
Layerwise Decay	0.95	0.95	0.95
Batch size	4	4	4
Loss function	Mean squared error	$\mathcal{L}_{\mathrm{HTS}}$ / $\mathcal{L}_{\mathrm{KL}}$	$\mathcal{L}^{\mathrm{E}}_{\mathrm{HTS}}$ [$\mathcal{L}^{\mathrm{E}}_{\mathrm{ABS}}$ / $\mathcal{L}^{\mathrm{E}}_{\mathrm{SQ}}$]
Dropout	0.15	0.15	0.15
Hidden sizes	[3072, 1024]	[3072, 1024]	[3072, 1024]
Encoder Embedding layer	Frozen	Frozen	Frozen
Bottleneck layer size	-	-	256
FP precision	32	32	32
No. Epochs (training)	2	2	2
No. Epochs (fine-tuning)	1	1	1

Table 7: Hyperparameters used to train uncertainty prediction methods.