Schroedinger's Threshold: When the AUC doesn't predict Accuracy

Juri Opitz
University of Zurich
opitz.sci@gmail.com

Abstract

The Area Under Curve measure (AUC) seems apt to evaluate and compare diverse models, possibly without calibration. An important example of AUC application is the evaluation and benchmarking of models that predict faithfulness of generated text. But we show that the AUC yields an academic and optimistic notion of accuracy that can misalign with the actual accuracy observed in application, yielding significant changes in benchmark rankings. To paint a more realistic picture of downstream model performance (and prepare a model for actual application), we explore different calibration modes, testing calibration data and method.

Keywords: Classification evaluation, AUC score, accuracy, calibration, faithfulness evaluation

1. Introcuction

In Natural Language Processing (NLP), we often want to compare diverse models in diverse domains and tasks. Consider Figure 1 that shows the answer of a dialog system to a user input. On the machine-generated output, we would like to use a model to judge whether the answer is faithful.[1] For this, we could draw from a huge shelf of models, including in/out-domain trained classifiers, or even metrics such as BERTscore (Zhang et al., 2020).

But how do we evaluate and compare such diverse models? When the target labels are *binary*, e.g., as they are indeed for text faithfulness (but also in many other NLP/ML tasks), it seems appealing to employ the Area Under Curve (AUC) measure. Indeed, AUC has a nice probabilistic interpretation and makes model calibration (i.e., searching for a decision threshold) unnecessary. Mainly for these reasons, the AUC has been explicitly recommended for evaluation and benchmarking of models that predict faithfulness (Honovich et al., 2022; Gekhman et al., 2023; Zha et al., 2023).

Yet, an issue is that AUC has an academic view on model power. In a "real-world" application, we cannot forgo model calibration, as we ultimately have to make decisions. In our example of text faithfulness, there are clear ramifications of different decision thresholds: with a false-positive we run a risk of releasing false or even harmful output; a false-negative may lead to censoring of good system output.

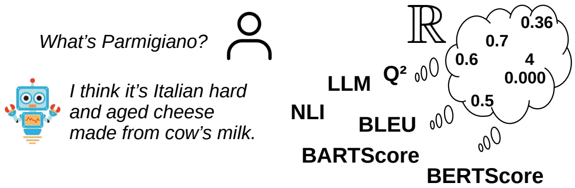

Figure 1: In NLP we witness diverse domains and tasks (here: dialog, faithfulness), and wonder about the predictive power of scores by diverse models (here: e.g., the BERT/BARTscore metric, task-focused systems such as the automatic Q/A metric 'Q^2' or Natural Language Inference systems, possibly also LLMs). While the AUC seems appealing as an assessment measure, it bears pitfalls.

In this paper, we show that such important real world considerations tend to be neglected by the AUC, and find that its theoretical perspective on system performance may not align with actual performance in applications. Our findings indicate that a main factor for this lies in the diversity of model score and data distributions. Indeed, we argue that AUC should not be used as a sole measure for model evaluation and benchmarking, particularly when models and data are diverse.

In sum, our main contribution is two-fold:

1. We show that the evaluation of diverse models with AUC can be misleading, and that AUC predicts mostly only the optimistic scenario of direct in-domain and in-distribution calibration.

2. We test different calibration strategies (varying development domain and method) for i) learning how to develop calibrated classifiers from diverse models and ii) best estimate their expected downstream classification performance.

Our code is available at `https://github.com/flipz357/SchroedingersEvaluation`.

[1] This particular task is well motivated: Today, text generation models produce millions of texts each day, and their output can still be unfaithful, with some assessing that LLM hallucination are inevitable (Xu et al., 2024). Thus, models that can reliably and efficiently assess faithfulness of generated text are of growing importance (Falke et al., 2019; Kryscinski et al., 2020; Wang et al., 2020; Maynez et al., 2020; Gekhman et al., 2023; Zha et al., 2023; Steen et al., 2023; Zhang et al., 2024).

LREC-COLING 2024, pages 14400–14406
May 20–25, 2024.

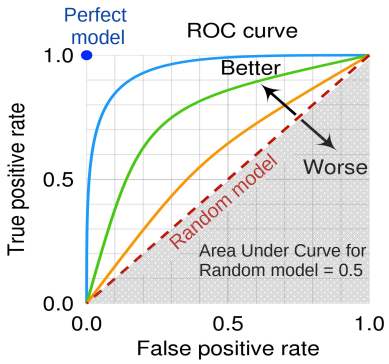

Figure 2: ROC curve examples of different models.

2. Preliminaries

AUC (or AUROC) is the *Area Under the Receiver Operating Characteristic Curve* (Fawcett, 2006). Given data $\{(x_i, y_i)\}_{i=1}^n$ with y_i a binary label and x_i an input mapped by a model to a score $s_i \in \mathbb{R}$, we can set threshold $\hat{\theta}$ to get a true positive rate $TPR(\hat{\theta})$ and false positive rate $FPR(\hat{\theta})$:

$$TPR(\hat{\theta}) = \frac{TP_{\hat{\theta}}}{TP_{\hat{\theta}} + FN_{\hat{\theta}}} \qquad FPR(\hat{\theta}) = \frac{FP_{\hat{\theta}}}{FP_{\hat{\theta}} + TN_{\hat{\theta}}}.$$

Given $I[c]$ returns 1 if the condition c is true, and 0 else, the $TP_{\hat{\theta}}$ is the amount of true positives $\sum_{i=1}^n I[s_i > \hat{\theta} \wedge y_i = 1]$; $TN_{\hat{\theta}}$ is the amount of true negatives $\sum_{i=1}^n I[s_i \leq \hat{\theta} \wedge y_i = 0]$; $FP_{\hat{\theta}}$ is the amount of false positives $\sum_{i=1}^n I[s_i > \hat{\theta} \wedge y_i = 0]$ and $FN_{\hat{\theta}}$ the amount of false negatives $\sum_{i=1}^n I[s_i \leq \hat{\theta} \wedge y_i = 1]$. With this, we can plot the receiver-operator curve (ROC) with TPR on the y-axis and FPR on the x-axis, and get the *area under curve* (AUC), which equals 1 for a perfect classifier and 0.5 for a random classifier (cf. Figure 2[2]).

The AUC score has an intuitive interpretation: Given two data instances with opposing labels, the AUC score tells us the probability that our model assigns a greater score to the positively labeled instance than to the instance with the negative label.

AUC seems appealing (theoretically): Besides its intuitive interpretation, the AUC score allows simple evaluation by factoring out model calibration (determining a threshold). Thus we can assess and compare seemingly fairly the *theoretical classification power* of diverse models such as metrics as well as non-calibrated classifiers (e.g., classifiers trained on different domains), and of course also standard classifiers that are already calibrated.

However, with this theoretic view on model power, the AUC makes us potentially neglect the final goal of most NLP systems: they should assign categorical decisions and show decision skill. If we'd presume that calibration of diverse models would be of same difficulty for any model, relying on AUC would perhaps seem fine. However, diverse models may return diverse score distributions. Data for finding a suitable threshold also can be diverse and noisy. Therefore we hypothesize that calibration suitability of models is also diverse, possibly affecting their real-world classification performance, with ramifications for the utility of AUC.

3. Experimental setup

Data sets are adopted from the popular TRUE benchmark (Honovich et al., 2022). TRUE combines a rich variety of faithfulness domains in a standardized format: summarization (Pagnoni et al., 2021; Maynez et al., 2020; Wang et al., 2020; Fabbri et al., 2021), knowledge-grounded dialog (Honovich et al., 2021; Gupta et al., 2022; Dziri et al., 2022), and paraphrases (Zhang et al., 2019).[3] TRUE explicitly recommends AUC evaluation.

Metrics that we include are BERTscore (Zhang et al., 2020) using either DeBERTa (He et al., 2020), henceforth denoted by `DBERTsc`, or RoBERTa (Liu et al., 2019), denoted by `RBERTsc`. As recommended by Honovich et al. (2022), we take their *precision* predictions, which should better assess faithfulness than F1 or recall. Then we also show `BARTsc`(ore) (Yuan et al., 2021), `BLEURT` (Sellam et al., 2020) and `BLEU` (k=4) (Papineni et al., 2002).

Models are also diverse. Some are NLI-based (a closely related task), while others employ elaborate scoring techniques, e.g., by analyzing a cross-product of sentences. As in TRUE, we employ `ANLI` (Honovich et al., 2022) which is a T5-11B (Raffel et al., 2020) LLM trained on ANLI (Nie et al., 2020). `SummacZS` (Laban et al., 2022) evaluates an NLI model on sentence pairs and averages maximum entailment probabilities, and `Q2` (Honovich et al., 2021) integrates a question-answering step.

3.1. Measurement of expected accuracy

Given are datasets $d_1, ..., d_n$ and a diverse model m that outputs a real number ('score'). It is intuitive to transform the score into a binary prediction by fitting a logistic curve with a bias β_0^m and a weight β_1^m, also known as *Platt scaling* (Platt et al., 1999):

$$p(x, m) = \frac{1}{1 + e^{-(\beta_0^m + \beta_1^m m(x))}} \tag{1}$$

With this, we can make a decision with natural probability threshold $\theta = 0.5$:

$$f(x, m) = \begin{cases} 1, & \text{if } p(x, m) > 0.5 \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

[2] Figure under public CC-BY-SA-4.0 license from public domain and further refined by the authors of this paper.

[3] Summarization: *quags, summeval, frank, quags-x, quags-c*. Dialog: *begin, dialfact, q2*. Paraphrase: *paws*.

data set	BLEU	QuestE	FactCC	SummaCC	SummacZS	BARTSc	RBERTSc	Q2	ANLI	DBERTSc	BLEURT
qags-c	63.9 | **11**	64.2 | **10**	76.4 | **6**	79.6 | **5**	80.9 | **3**	80.9 | **4**	74.8 | **7**	83.5 | **1**	82.1 | **2**	69.1 | **9**	71.6 | **8**
summeval	60.2 | **11**	70.1 | **9**	75.9 | **6**	79.8 | **3**	81.7 | **1**	73.5 | **7**	73.0 | **8**	78.8 | **4**	80.5 | **2**	77.2 | **5**	66.7 | **10**
frank	78.0 | **10**	84.0 | **7**	76.4 | **11**	88.9 | **3**	89.1 | **2**	86.1 | **5**	80.8 | **9**	87.8 | **4**	89.4 | **1**	84.3 | **6**	82.8 | **8**
qags-x	48.6 | **11**	56.3 | **7**	64.9 | **5**	76.1 | **3**	78.1 | **2**	53.8 | **8**	52.8 | **9**	70.9 | **4**	83.8 | **1**	49.5 | **10**	57.2 | **6**
dialfact	72.5 | **7**	77.3 | **5**	55.3 | **11**	81.2 | **3**	84.1 | **2**	65.6 | **8**	62.9 | **10**	86.1 | **1**	77.7 | **4**	64.2 | **9**	73.1 | **6**
mnbm	49.3 | **11**	65.3 | **6**	59.4 | **10**	67.2 | **4**	71.3 | **2**	60.9 | **9**	65.5 | **5**	68.7 | **3**	77.9 | **1**	62.8 | **8**	64.5 | **7**
begin	84.6 | **5**	84.1 | **6**	64.4 | **11**	81.6 | **9**	82.0 | **8**	86.3 | **4**	87.1 | **2**	79.7 | **10**	82.6 | **7**	87.9 | **1**	86.4 | **3**
q2	64.3 | **10**	72.2 | **6**	63.7 | **11**	77.5 | **2**	77.4 | **3**	64.9 | **8**	64.8 | **9**	80.9 | **1**	72.7 | **4**	70.0 | **7**	72.4 | **5**
paws	77.3 | **7**	69.2 | **9**	64.0 | **11**	88.2 | **2**	88.2 | **3**	77.5 | **5**	69.3 | **8**	89.7 | **1**	86.4 | **4**	77.5 | **6**	68.3 | **10**
mean	66.5 | **11**	71.4 | **7**	66.7 | **10**	80.0 | **4**	81.4 | **2**	72.2 | **5**	70.1 | **9**	80.7 | **3**	81.5 | **1**	71.4 | **8**	71.4 | **6**

Table 1: AUC evaluation (x100).

data set	BLEU	QuestE	FactCC	SummaCC	SummacZS	BARTSc	RBERTSc	Q2	ANLI	DBERTSc	BLEURT
qags-c	51.9 | **9**	58.3 | **8**	51.9 | **10**	59.6 | **7**	65.5 | **4**	68.9 | **1**	68.5 | **2**	63.8 | **5**	67.7 | **3**	60.0 | **6**	51.9 | **11**
summeval	18.5 | **9**	51.8 | **5**	18.4 | **10**	82.9 | **3**	82.6 | **4**	22.0 | **6**	19.1 | **8**	86.2 | **1**	85.5 | **2**	21.3 | **7**	18.4 | **11**
frank	66.8 | **9**	73.3 | **7**	66.8 | **10**	79.4 | **1**	74.1 | **6**	68.6 | **8**	77.3 | **4**	78.8 | **2**	74.7 | **5**	77.5 | **3**	66.8 | **11**
qags-x	51.5 | **7**	51.5 | **8**	51.5 | **9**	61.1 | **4**	69.0 | **2**	52.3 | **6**	53.1 | **5**	62.3 | **3**	75.7 | **1**	51.0 | **11**	51.5 | **10**
dialfact	62.0 | **6**	66.1 | **5**	56.2 | **11**	66.3 | **4**	69.3 | **3**	61.4 | **9**	61.0 | **10**	74.4 | **1**	70.4 | **2**	61.5 | **8**	61.7 | **7**
mnbm	89.8 | **2**	89.8 | **3**	88.2 | **7**	89.4 | **5**	88.8 | **6**	89.8 | **1**	87.9 | **8**	86.7 | **9**	73.6 | **10**	64.2 | **11**	89.8 | **4**
begin	74.5 | **8**	76.2 | **6**	70.8 | **11**	76.9 | **5**	80.3 | **1**	72.2 | **10**	79.9 | **2**	76.0 | **7**	79.1 | **3**	78.9 | **4**	72.5 | **9**
q2	44.0 | **9**	53.1 | **5**	42.3 | **11**	59.6 | **4**	63.9 | **2**	42.4 | **10**	45.6 | **7**	73.2 | **1**	60.5 | **3**	48.3 | **6**	44.1 | **8**
paws	50.6 | **8**	49.9 | **9**	53.0 | **7**	80.7 | **1**	69.2 | **4**	46.4 | **10**	53.7 | **6**	73.9 | **3**	78.9 | **2**	67.4 | **5**	44.4 | **11**
mean	56.6 | **9**	63.3 | **5**	55.4 | **11**	72.9 | **4**	73.6 | **3**	58.2 | **8**	60.7 | **6**	75.0 | **1**	74.0 | **2**	58.9 | **7**	55.7 | **10**

Table 2: Expected accuracy evaluation (x100).

So calibrating our model m means finding suitable β_0^m, β_1^m. To calculate the generalization accuracy of m, it is intuitive to adopt the following strategy: For any unseen testing data set d_i, we calibrate Eq. 1, by tuning β_0^m, β_1^m on all $d_{j \neq i}$. Finally, we get the expected accuracy on our testing data set d_i:

$$acc(d_i) = \frac{\sum_{(x,y) \in d_i} I[f(x,m) = y]}{|d_i|}. \qquad (3)$$

Note that in contrast to AUC, our expected accuracy measurement is real-world oriented: Assume we have a metric such as BERTScore (Zhang et al., 2020) – how would an applicant transform this metric into a faithfulness predictor for filtering their generation system output? Clearly, they would need to perform calibration using development data. With our setup, we simulate this important scenario and obtain an expected accuracy score.

4. AUC mispredicts accuracy

4.1. Experiment goal

The main goal of our experiment is to investigate our hypothesis that AUC can yield a wrong picture about actual performance of models. To this aim, we conduct a real-world oriented downstream task simulation of diverse faithfulness models, measuring their expected accuracy (as detailed above).

4.2. Experiment results

We compare Table 1 (AUC of models) against Table 2 (expected accuracy). Interestingly, changes are more drastic than we had initially suspected. In fact, they even result in a **change of the best system on the benchmark**: the Q/A based system `Q2` ranks third after `ANLI` and `SummacZS` in average AUC, but according to the average accuracy, it obtains rank 1 (an improvement of two ranks). Then we also observe **interesting cases of ranking changes** of other metrics: for instance, `BLEU` yields a low rank according to AUC in the mnbm data set (rank 11), but performs much better accuracy-wise (rank 2).

4.3. Studying score distribution

We saw that AUC may not predict estimated downstream accuracy. But why would some models be more negatively/positively affected by calibration? A reason may lie in the models' score distribution and their suitability for calibration. Therefore, we investigate the models' empirical distributions.

Why would `Q2` be preferable over `ANLI`? This question is interesting, since we saw that the best performing models differ between AUC and expected downstream accuracy. The two models are also diverse, since `Q2` employs a Q/A module while `ANLI` is an LLM trained on NLI. Their histograms (Figure 3) differ much: while both `ANLI` and `Q2` tend to the extremes of the spectrum, the effect is much more pronounced for `ANLI`. Throughout the scale, `Q2` appears to be more 'balanced'. For the `ANLI` distribution, the data already seems harshly discriminated in two classes, perhaps increasing the difficulty of finding a generalizable threshold.

Less variance → easier calibration? We create two groups of models: those that obtain a better

	metric names and variance						mean
better	B	Q	R	Q2	D		
metrics	0.03	0.02	0.02	0.14	0.02		0.05
worse	F	SC	SZ	BA	A	BL	
metrics	0.16	0.08	0.20	0.01	0.24	0.03	0.12

Table 3: Variance of metric scores that perform better/worse under expected accuracy

.

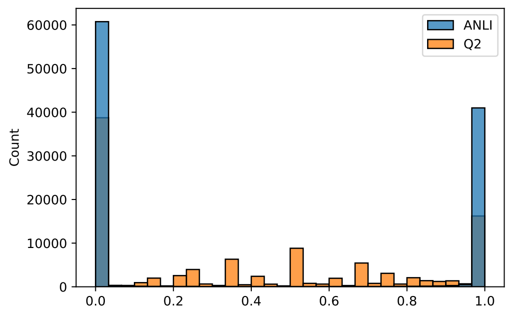

Figure 3: Histograms of best performing models `Q2` and `ANLI`. `Q2` performs best according to expected accuracy, `ANLI` performs best according to AUC.

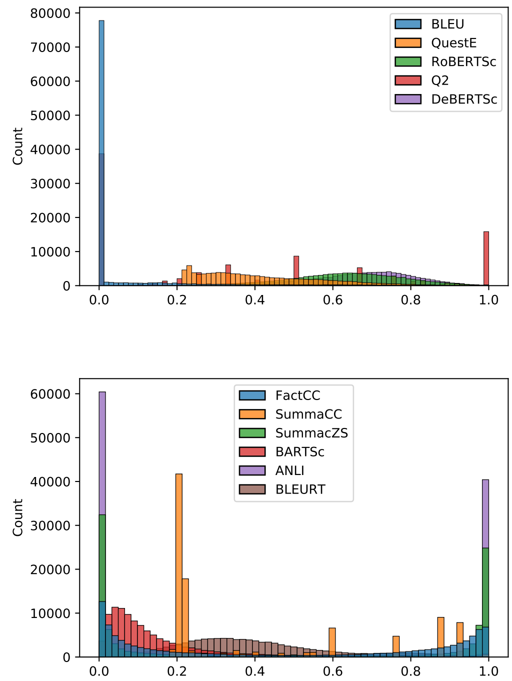

Figure 4: **Top**: histograms of models that perform *better* under expected accuracy (vs. AUC). Bottom: histograms of models that perform *worse*.

rank according to accuracy, and those that obtain a worse rank. The histograms are shown in Figure 4. We see that models that are relatively more negatively affected by calibration tend to show more skewed distributions. The scores of the better models, however, seem more balanced and also exhibit a smaller average variance (Table 3: avg. variance of better metrics=0.05; avg. variance of worse metrics=0.12).

5. Analysis

5.1. Effect of calibration technique

We want to study the effects of different approaches to calibration. The diversity of models and data in TRUE provides an interesting study environment. Our first setup is aimed at testing the classification performance in dependence of the nature of the training data. This lets us assess domain effects and generalization power as calibration effects. For the second setup we investigate different calibration algorithms, to shed more light on the question: How to best transform a diverse model into a faithfulness assessment?

Setup I: Domain Effects & Generalization. We denote the cross-domain setup from the section before as *Xdomain*. Additionally, we introduce the arguably hard setup of *OutDomain* which is interesting since it only allows training on out-domain training data and thus tests the transfer to new domains. Other setups are *InDomain* that allows calibration only in in-domain training data, and *InData*, where we have in-domain and in-task training data, where we would naturally expect the best although much less generalizable performance. For *InData*, we estimate performance on a random 80/20 train/test split of a data set, averaged over 100 repetitions.

Setup II: Calibration method effects. The intuitive logistic curve calibration is by far not the only possible calibration method. In fact, it has also been criticized (Silva Filho et al., 2023), e.g., due to observed over-confidence effects.

To test another method of probabilistic calibration, we run experiments with Isotonic regression (Niculescu-Mizil and Caruana, 2005). To every training datum (x_i, y_i), *isotonic* finds a $\hat{y}_i$ s.t. $(y_i - \hat{y}_i)^2$ is minimized and $\forall j : x_j \geq x_i \implies \hat{y}_j \geq \hat{y}_i$. Prediction of an unseen datum x_k is then performed through interpolation: $\hat{y}_k = \hat{y}_l + \frac{x_k - x_l}{x_r - x_l}(\hat{y}_r - \hat{y}_l)$ if $x_l \leq x_k \leq x_r$, and else either $\hat{y}_l$ (if $x_k < x_l$) or $\hat{y}_r$ (if $x_k > x_r$). Additionally, we test a non-probabilistic $(\theta \neq 0.5)$ method of *decision stump* that is a decision tree with depth=1, searching for one threshold that empirically best divides the training data.

Results. Table 4 shows the mean (over each data set) accuracy results for variations of calibration method and variations of calibration data. We make

metric	BLEU	QuestE	FactCC	SummaCC	SummacZS	BARTSc	RBERTSc	Q2	ANLI	DBERTSc	BLEURT	AVG
AUC	66.5 | **11**	71.4 | **7**	66.7 | **10**	80.0 | **4**	81.4 | **2**	72.2 | **5**	70.1 | **9**	80.7 | **3**	81.5 | **1**	71.4 | **8**	71.4 | **6**	73.9
Xdomain	56.6 | **9**	63.3 | **5**	55.4 | **11**	**72.9** | **4**	**73.6** | **3**	**58.2** | **8**	**60.7** | **6**	75.0 | **1**	74.0 | **2**	58.9 | **7**	55.7 | **10**	64.0
-Isotonic	55.5 | **10**	65.0 | **5**	**62.7** | **6**	70.2 | **4**	71.4 | **3**	57.2 | **9**	58.8 | **8**	**75.7** | **1**	74.0 | **2**	**59.2** | **7**	55.0 | **11**	64.1
-stump	**57.6** | **9**	**68.0** | **5**	59.3 | **6**	70.9 | **3**	70.6 | **4**	56.8 | **11**	57.6 | **8**	**75.7** | **1**	**74.1** | **2**	58.0 | **7**	**56.9** | **10**	64.1
OutDomain	**56.3** | **10**	62.8 | **5**	56.7 | **9**	**72.3** | **4**	**73.3** | **3**	**58.4** | **8**	**60.5** | **7**	74.3 | **1**	**73.8** | **2**	**60.7** | **6**	**55.1** | **11**	64.0
-Isotonic	55.5 | **10**	63.0 | **5**	60.7 | **6**	63.8 | **4**	71.9 | **2**	57.2 | **9**	58.3 | **8**	74.7 | **1**	71.6 | **3**	59.6 | **7**	55.0 | **11**	62.8
-stump	55.4 | **10**	**63.3** | **5**	**62.7** | **6**	64.1 | **4**	69.1 | **3**	56.8 | **9**	57.6 | **8**	**75.2** | **1**	71.7 | **2**	58.1 | **7**	54.7 | **11**	62.6
OutData	54.3 | **11**	62.8 | **5**	54.8 | **10**	73.9 | **2**	73.9 | **3**	55.7 | **8**	56.1 | **7**	74.7 | **1**	72.5 | **4**	56.4 | **6**	55.5 | **9**	62.8
-Isotonic	57.9 | **6**	61.4 | **5**	55.5 | **9**	67.7 | **4**	71.4 | **2**	55.3 | **10**	56.2 | **8**	73.9 | **1**	69.0 | **3**	54.4 | **11**	56.8 | **7**	61.7
-stump	56.2 | **7**	60.7 | **5**	55.4 | **10**	65.2 | **4**	67.2 | **2**	54.6 | **11**	55.7 | **9**	73.5 | **1**	67.0 | **3**	56.0 | **8**	56.6 | **6**	60.7
InDomain***	51.2 | **10**	69.7 | **4**	57.3 | **8**	74.9 | **3**	75.7 | **2**	61.8 | **7**	**50.8** | **11**	77.0 | **1**	66.9 | **5**	**51.9** | **9**	64.2 | **6**	63.8
-Isotonic***	**65.1** | **8**	**70.6** | **5**	61.4 | **9**	75.3 | **4**	77.0 | **2**	65.4 | **7**	48.8 | **11**	77.3 | **1**	76.5 | **3**	51.0 | **10**	67.0 | **6**	66.9
-stump***	61.2 | **9**	70.4 | **5**	**61.9** | **8**	**75.6** | **4**	76.8 | **3**	**66.4** | **7**	48.4 | **10**	**77.6** | **2**	**77.7** | **1**	47.0 | **11**	**69.3** | **6**	66.6
InData	66.9 | **11**	70.1 | **8**	69.1 | **10**	75.5 | **4**	77.7 | **2**	70.2 | **7**	69.6 | **9**	79.3 | **1**	75.7 | **3**	70.8 | **5**	70.3 | **6**	72.3
-Isotonic	**69.9** | **11**	**71.3** | **7**	**70.3** | **10**	**77.6** | **4**	**78.2** | **3**	**72.7** | **5**	**70.8** | **9**	78.4 | **1**	**78.4** | **2**	**71.0** | **8**	**71.9** | **6**	73.7
-stump	69.6 | **9**	70.3 | **8**	69.4 | **11**	76.7 | **4**	77.4 | **3**	71.2 | **6**	69.4 | **10**	**78.5** | **1**	78.4 | **2**	70.3 | **7**	71.3 | **5**	73.0

Table 4: Different modes of calibration, varying calibration method and training data. Mean performance over all data sets. In each group of three lines: The first line is calibration via *logistic regression*, the second line is *isotonic regression*, and the third is *decision stump*. Note the assumptions on data availibility: *Indata* requires annotated in-domain in-task training; *InDomain* needs in-domain training; *XDomain* lessens this dependence, and *OutDomain* is the most general setup. ***: A data subset (PAWS) is calibrated InData (instead of Indomain), since it is the only data set of domain *paraphrase/wiki*.

metric	BLEU	QuestE	FactCC	SummaCC	SummacZS	BARTSc	RBERTSc	Q2	ANLI	DBERTSc	BLEURT	AVG
AUC	66.5 | **11**	71.4 | **7**	66.7 | **10**	80.0 | **4**	81.4 | **2**	72.2 | **5**	70.1 | **9**	80.7 | **3**	81.5 | **1**	71.4 | **8**	71.4 | **6**	73.9
κ, XDomain	12.7 | **10**	23.1 | **5**	14.7 | **9**	31.4 | **4**	34.5 | **3**	8.8 | **11**	18.9 | **7**	40.7 | **1**	40.7 | **2**	19.3 | **6**	18.9 | **8**	24.0
κ, OutDomain	4.3 | **10**	17.7 | **6**	13.7 | **8**	30.1 | **4**	33.9 | **3**	8.5 | **9**	17.1 | **7**	39.8 | **1**	39.6 | **2**	19.3 | **5**	1.4 | **11**	20.5
κ, Outdata	14.7 | **7**	23.2 | **5**	10.7 | **10**	42.8 | **3**	45.1 | **2**	10.5 | **8**	11.3 | **9**	46.4 | **1**	41.1 | **4**	9.8 | **11**	15.0 | **6**	24.6
κ, Indomain	21.4 | **8**	28.3 | **6**	18.1 | **10**	37.9 | **4**	42.7 | **3**	25.1 | **7**	18.2 | **9**	45.1 | **2**	46.8 | **1**	14.9 | **11**	29.7 | **5**	29.8
κ, Indata	21.5 | **10**	25.2 | **8**	17.3 | **11**	39.9 | **4**	41.4 | **3**	28.9 | **5**	24.3 | **9**	45.1 | **1**	42.1 | **2**	27.1 | **6**	26.0 | **7**	30.8

Table 5: Evaluation with KAPPA (κ) after calibration reveals the hardness of predicting faithfulness. For each model/column and calibration data/row, the best score over three calibration methods is displayed.

some observations: i) As expected, *InData* is the easiest setup, yielding highest accuracy (up to 73.7 accuracy with isotonic calibration). ii) Out-domain generalized calibration is hard. Here Logistic calibration provides overall best calibration (64.0 accuracy). iii) Again, there is no ranking that is same as under AUC, and all calibrated accuracy scores tend to be much lower thatn AUC. iv) different calibration methods can yield different results, but we cannot make generalizing statement as to which calibration method would be overall preferable.

Notably, only in the easy and strongly data-dependent setup of *InData* calibration, AUC somewhat aligns with the expected accuracy. The relatively high scores and easiness of this setup suggest that AUC is an *optimistic* performance measure, especially when data and models are diverse.

5.2. Other classification metrics

Calibrated classifiers can be evaluated with metrics other than accuracy. We show the KAPPA score as a chance-corrected accuracy measure with a random baseline score of 0.0, correcting for label skew (Opitz, 2024). Results in Table 5 reveal the hardness of the task: Many measures are not much better than the chance baseline, even the best observed KAPPA score still seem low.

6. Conclusions

When evaluating diverse models as binary classifiers, it seems appealing to use the AUC score for benchmarking and evaluation (specifically since it factors out calibration). But we show that AUC may fail to predict the accuracy that can be expected in an application. Our work can be both interpreted as a warning to not rely (only) on AUC for evaluation as well as a call for reflecting on application when evaluating diverse *decision* models.

7. Acknowledgments

We thank our anonymous reviewers B, C and D for their constructive comments that have improved this paper, and the LREC-COLING PC chairs for a helpful communication.

8. Bibliographical References

Nouha Dziri, Hannah Rashkin, Tal Linzen, and David Reitter. 2022. Evaluating Attribution in Dialogue Systems: The BEGIN Benchmark. *Transactions of the Association for Computational Linguistics*, 10:1066–1083.

Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. SummEval: Re-evaluating summarization evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Tobias Falke, Leonardo F. R. Ribeiro, Prasetya Ajie Utama, Ido Dagan, and Iryna Gurevych. 2019. Ranking generated summaries by correctness: An interesting but challenging application for natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2214–2220, Florence, Italy. Association for Computational Linguistics.

Tom Fawcett. 2006. An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874.

Zorik Gekhman, Jonathan Herzig, Roee Aharoni, Chen Elkind, and Idan Szpektor. 2023. Trueteacher: Learning factual consistency evaluation with large language models. *arXiv preprint arXiv:2305.11171*.

Prakhar Gupta, Chien-Sheng Wu, Wenhao Liu, and Caiming Xiong. 2022. DialFact: A benchmark for fact-checking in dialogue. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3785–3801, Dublin, Ireland. Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. 2022. TRUE: Re-evaluating factual consistency evaluation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3905–3920, Seattle, United States. Association for Computational Linguistics.

Or Honovich, Leshem Choshen, Roee Aharoni, Ella Neeman, Idan Szpektor, and Omri Abend. 2021. q^2: Evaluating factual consistency in knowledge-grounded dialogues via question generation and question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7856–7870, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online. Association for Computational Linguistics.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online. Association for Computational Linguistics.

Alexandru Niculescu-Mizil and Rich Caruana. 2005. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, page 625–632, New York, NY, USA. Association for Computing Machinery.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Juri Opitz. 2024. A Closer Look at Classification Evaluation Metrics and a Critical Reflection of Common Evaluation Practice. *TACL (to appear)*.

Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. 2021. Understanding factuality in abstractive summarization with FRANK: A benchmark for factuality metrics. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4812–4829, Online. Association for Computational Linguistics.

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.

John Platt et al. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. BLEURT: learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7881–7892. Association for Computational Linguistics.

Telmo Silva Filho, Hao Song, Miquel Perello-Nieto, Raul Santos-Rodriguez, Meelis Kull, and Peter Flach. 2023. Classifier calibration: a survey on how to assess and improve predicted class probabilities. *Mach. Learn.*, 112(9):3211–3260.

Julius Steen, Juri Opitz, Anette Frank, and Katja Markert. 2023. With a little push, NLI models can robustly and efficiently predict faithfulness. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 914–924, Toronto, Canada. Association for Computational Linguistics.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online. Association for Computational Linguistics.

Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. 2024. Hallucination is inevitable: An innate limitation of large language models. *arXiv preprint arXiv:2401.11817*.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 27263–27277.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. *arXiv preprint arXiv:2305.16739*.

Huajian Zhang, Yumo Xu, and Laura Perez-Beltrachini. 2024. Fine-grained natural language inference based faithfulness evaluation for diverse summarisation tasks. In *arxiv, EACL 2024 (to appear)*. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: Paraphrase adversaries from word scrambling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1298–1308, Minneapolis, Minnesota. Association for Computational Linguistics.