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. Introduction

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.

1This 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; Krzycinski 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).

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.
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 \(FPR(\hat{\theta})\):

\[
TPR(\hat{\theta}) = \frac{TP_{\hat{\theta}}}{TP_{\hat{\theta}} + FN_{\hat{\theta}}}, \quad FPR(\hat{\theta}) = \frac{FP_{\hat{\theta}}}{FP_{\hat{\theta}} + TN_{\hat{\theta}}}.
\]

Given \(I[c]\) returns 1 if the condition \(c\) is true, and 0 else, the \(TP_{\hat{\theta}}\) is the amount of true positives \(\sum_{i=1}^n I[s_i > \hat{\theta} \land y_i = 1]\); \(TN_{\hat{\theta}}\) is the amount of true negatives \(\sum_{i=1}^n I[s_i \leq \hat{\theta} \land y_i = 0]\); \(FP_{\hat{\theta}}\) is the amount of false positives \(\sum_{i=1}^n I[s_i > \hat{\theta} \land y_i = 0]\); and \(FN_{\hat{\theta}}\) the amount of false negatives \(\sum_{i=1}^n I[s_i \leq \hat{\theta} \land y_i = 1]\). 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).

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). 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 DeBERTsc 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 (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 \(d_1, \ldots, d_n\) 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 \(\beta_m^0\) and a weight \(\beta_m^1\), also known as Platt scaling (Platt et al., 1999):

\[
p(x, m) = \frac{1}{1 + e^{-(\beta_m^0 + \beta_m^1 m(x))}}
\]

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

\[
f(x, m) = \begin{cases} 1, & \text{if } p(x, m) > 0.5 \\ 0, & \text{otherwise} \end{cases}
\]

---

\(^2\)Figure under public CC-BY-SA-4.0 license from public domain and further refined by the authors of this paper.

data set BLEU QuestE FactCC SummaCC SummacZS BARTSc RBERTSc Q2 ANLI DBERTSc BLEURT
qags-c 63.9 | 11 64.2 | 10 76.4 | 6 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 | 5 79.8 | 3 81.7 | 1 73.5 | 7 73.0 | 8 78.8 | 4 80.5 | 0 85.2 | 5 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
diifact 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 | 6 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).

<table>
<thead>
<tr>
<th>data set</th>
<th>BLEU</th>
<th>QuestE</th>
<th>FactCC</th>
<th>SummaCC</th>
<th>SummacZS</th>
<th>BARTSc</th>
<th>RBERTSc</th>
<th>Q2</th>
<th>ANLI</th>
<th>DBERTSc</th>
<th>BLEURT</th>
</tr>
</thead>
<tbody>
<tr>
<td>qags-c</td>
<td>63.9</td>
<td>11</td>
<td>64.2</td>
<td>10</td>
<td>76.4</td>
<td>6</td>
<td>80.9</td>
<td>3</td>
<td>80.9</td>
<td>4</td>
<td>74.8</td>
</tr>
<tr>
<td>summeval</td>
<td>60.2</td>
<td>11</td>
<td>70.1</td>
<td>9</td>
<td>75.9</td>
<td>5</td>
<td>79.8</td>
<td>3</td>
<td>81.7</td>
<td>1</td>
<td>73.5</td>
</tr>
<tr>
<td>frank</td>
<td>78.0</td>
<td>10</td>
<td>84.0</td>
<td>7</td>
<td>76.4</td>
<td>11</td>
<td>88.9</td>
<td>3</td>
<td>89.1</td>
<td>2</td>
<td>86.1</td>
</tr>
<tr>
<td>qags-x</td>
<td>48.6</td>
<td>11</td>
<td>56.3</td>
<td>7</td>
<td>64.9</td>
<td>5</td>
<td>76.1</td>
<td>3</td>
<td>78.1</td>
<td>2</td>
<td>53.8</td>
</tr>
<tr>
<td>diifact</td>
<td>72.5</td>
<td>7</td>
<td>77.3</td>
<td>5</td>
<td>55.3</td>
<td>11</td>
<td>81.2</td>
<td>3</td>
<td>84.1</td>
<td>2</td>
<td>65.6</td>
</tr>
<tr>
<td>mnbm</td>
<td>49.3</td>
<td>11</td>
<td>65.3</td>
<td>6</td>
<td>59.4</td>
<td>10</td>
<td>67.2</td>
<td>4</td>
<td>71.3</td>
<td>2</td>
<td>60.9</td>
</tr>
<tr>
<td>begin</td>
<td>84.6</td>
<td>5</td>
<td>84.1</td>
<td>6</td>
<td>64.4</td>
<td>11</td>
<td>81.6</td>
<td>9</td>
<td>82.0</td>
<td>8</td>
<td>86.3</td>
</tr>
<tr>
<td>q2</td>
<td>64.3</td>
<td>10</td>
<td>72.2</td>
<td>6</td>
<td>63.7</td>
<td>11</td>
<td>77.5</td>
<td>2</td>
<td>77.4</td>
<td>3</td>
<td>64.9</td>
</tr>
<tr>
<td>paws</td>
<td>77.3</td>
<td>7</td>
<td>69.2</td>
<td>9</td>
<td>64.0</td>
<td>11</td>
<td>88.2</td>
<td>2</td>
<td>88.2</td>
<td>3</td>
<td>77.5</td>
</tr>
<tr>
<td>mean</td>
<td>66.5</td>
<td>11</td>
<td>71.4</td>
<td>7</td>
<td>66.7</td>
<td>10</td>
<td>80.0</td>
<td>4</td>
<td>81.4</td>
<td>2</td>
<td>72.2</td>
</tr>
</tbody>
</table>

Table 2: Expected accuracy evaluation (x100).

So calibrating our model $m$ means finding suitable $\beta_m^0$, $\beta_m^1$. 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 $\beta_m^0$, $\beta_m^1$ on all $d_{i\neq i}$. Finally, we get the expected accuracy on our testing data set $d_i$:

$$\text{acc}(d_i) = \frac{\sum_{(x,y) \in d_i} I[f(x, m) = y]}{|d_i|}.$$ (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 agent 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...
metrics and variance
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 ... accuracy results for variations of calibration
method and variation of calibration data. We make

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.

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 interest-
ing since it only allows training on out-domain train-
ing data and thus tests the transfer to new domains.

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

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 calibra-
tion, 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: \(\hat{y}_k = \hat{y}_i + \frac{x_k - x_i}{x_r - x_i} (\hat{y}_r - \hat{y}_i)\) if
\(x_l \leq x_k \leq x_r\), and else either \(\hat{y}_i\) (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 deci-
sion 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
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 than 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 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 Dia-


