Applying Calibration Methods to NLP Models

Matthew Jimenez, Quentin Mot, Jack York
Georgia Institute of Technology

Abstract

In 2017, contemporary out of the box neural networks were poorly calibrated despite being more accurate than their older counterparts (Guo et al., 2017). To the best of our knowledge, no one has yet replicated the authors' experiments on Large Language Models. We replicate histogram binning, isotonic regression, Platt scaling, matrix scaling, and temperature scaling to LUKE, T5, and ACE. We show that LLMs continue the trend of deeper models trained on more data being even more poorly calibrated than their ancestors, with uncalibrated Expected Calibration Error much higher than the previously examined ResNet, DenseNet, LeNet, DAN, and LSTM models (Guo et al., 2017). Moreover, we also find that Matrix scaling best calibrates LLM logits for the classification tasks.

1 Introduction

Dropout layers, residual connections, increasing model depth, pretraining on large datasets, and other innovations have drastically increased model accuracy on classification tasks (Simonyan and Zisserman, 2015) (He et al., 2015) (Huang et al., 2018). But, those architecture changes also decreased calibration on the same tasks (Guo et al., 2017). Model calibration, which we informally define as how likely the model's classification is correct, is quite important. For example, consider a self-driving car performing scene classification on pedestrians. An accurate model can often avoid misclassification. But, at least in its naive implementation, it cannot learn to defer to other sensors, behaviors, or systems in the case of low confidence.

Since the release of ChatGPT, Large Language Models (LLMs) are increasingly used for Natural Language Inference (NLI), Question Answering, Knowledge Distillation, and other applications. To consider yet another example, suppose that you ask chatGPT a question (Naveed et al., 2023). Clearly, it would be useful to know how likely ChatGPT thinks the answer is correct. Hence, evaluating and proposing improvement to LLM calibration is useful for researchers, developers, and end users.

2 Definitions

2.1 Confidence

Confidence is a measure of the average supposed probability of the predicted class within a set of outputs. When measuring confidence and calibration error, one separates the probabilities p into several (typically between 15-20) equally spaced bins. Within each bin, we calculate the arithmetic mean of the probabilities within our bin in order to calculate the confidence of said bin. This confidence can be compared against the accuracy of that very same bin.

2.2 Expected Calibration Error (ECE)

In a perfect model, the accuracy and confidence scores would be identical. That is:

$$\mathbb{E}_{\hat{P}}[|\mathbb{P}(\hat{Y} = Y|\hat{P} = p) - p|]$$

Expected Calibration Error (ECE) gives us a measure of the difference between the accuracy and confidence of our outputs, giving us a loss function to minimize within our calibration methods. It is defined as follows:

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)|$$

Where the bin B_m is the set of indices whose confidences fall in the range $(\frac{m-1}{M}, \frac{m}{M}]$. Essentially, this is a weighted mean of the (absolute value) difference between each bin's accuracy and confidence, with the weights being the number of entries within each bin. Because ECE is a measure of the difference between confidence and accuracy (essentially,

a measure of error) this is the figure we aim to minimize in the following experiments.

3 Calibration Methods

The calibration methods utilized in our experiments are intended to minimize a model's Expected Calibration Error (ECE)

3.1 Histogram Binning

Histogram Binning is a non-parametric calibration method that assigns all uncalibrated predictions $\hat{p}_i$ into mutually exclusive bins $B_1, ..., B_m$ where m is a hyperparameter. The bins partition $[0, 1]$ either such that each bin has equal length or that each bin contains an equal number of samples. At test time, uncalibrated predictions are turned into calibrated predictions by assigning all uncalibrated predictions in the interval of B_i the calibrated prediction θ_i, where θ_i can be the maximum value in the interval, the minimum value in the interval, a hyperparameter, or the solution that minimizes the bin-wise square loss, depicted below:

$$\min \sum_{m=1}^{M} \sum_{i=1}^{n} \mathbb{1}(a_m \leq \hat{p}_i < a_{m+1})(\theta_m - y_i)^2$$

3.2 Isotonic Regression

Isotonic Regression in another non-parametric calibration method that estimates a piecewise-constant function f which maps uncalibrated inputs within an interval to a single value. Formally, isotonic regression solves the following optimization problem:

$$\min_S \sum_{m=1}^{M} \sum_{i=1}^{n} \mathbb{1}(a_m \leq \hat{p}_i < a_{m+1}(\theta_m - y_i)^2)$$

$$S = \{M, \theta_1, ..., \theta_M, a_1, ..., a_{M+1}\}$$

As defined above Isotonic Regression is a generalization of histogram binning which jointly optimizes each bin's prediction parameter and the boundaries of each bin instead of solely each bin's prediction parameter.

3.3 Matrix Scaling

Histogram Binning and Isotonic Regression both apply best to binary classification. Matrix scaling is the first method we used specifically designed for mutli-class classification. Matrix scaling learns a one-versus-all linear classifier on the logits of a model. Formally, let $\mathbf{z}$ be a model's logits. Then, Matrix scaling learns parameters W_k and $\mathbf{b}_k$ for each of the k classes such that the unnormalized but calibrated probability vector $q^{(i)}$ for each input x_i equals $(\hat{q_1}, ..., \hat{q_k})$ where $\hat{q_j} := g(W_j\mathbf{z} + \mathbf{b}_j)$ for each class $j \in \{1, .., k\}$ where g is a specified nonlinearity. Then the confidence of the prediction is a normalized maximum entry of $q^{(i)}$, $\frac{max_k(q_j)}{\sum_{l=1}^{k} \hat{q_l}}$ and the prediction is the argmax of the same vector $q^{(i)}$.

3.4 Temperature Scaling

Temperature Scaling is similar to the temperature implementation of Project 3 but with the temperature T being a scalar parameter strictly greater than zero optimized on the training set for a classification task. Formally, for an input x_i with logits $\mathbf{z}_i$ let the calibrated logits vector $\hat{q_i} := g(\mathbf{z}_i/T)$. Then, similar to Matrix Scaling the confidence is the maximum entry of the calibated logits vector and the class prediction is the argmax of the same vector.

4 Experiments

4.1 Calibration on NER Outputs (LUKE)

The Language Understanding with Knowledge-based Embeddings model (LUKE) is an entity-typing model, in our case used for the common NLP problem of Named-Entity Recognition (NER). (Ikuya Yamada et al., 2020)

The logits outputted from LUKE are not representations of the possible classifications of each word, but instead possible classifications of a particular sequential span of words within a sentence. (Ikuya Yamada et al., 2020) This tackles the issue of named entities often spanning several words - with no clear limit as to how long an entity can be. Each element of the output represents every possible contiguous span that can be formed from the input sentence. The model then greedily selects from the highest values of the output logits within these spans in order to classify each word in the sentence as a particular named-entity (or simply as a non-named-entity). In essence, the spans with lower maximum probabilities are thrown out in favor of those with a higher estimated likelihood.

Our post-processing steps of calibration require operation on the logits (pre-activation layer values), the final probabilities, as well as the gold labels given by the dataset. With our logits / probabilities

being represented on a *span* level, while the gold labels only provide values on a *word-by-word* level, there has to be some degree of compromise in terms of how the logits are chosen - a many-to-one operation. We chose to use the very same logits that the model greedily selects for each word in post-processing, discarding other logits as described previously. To an extent, this makes calibration slightly less effective seeing as we cannot operate on all available logits - that being said, a compromise must be made here in order to match logits to gold labels in a one-to-one fashion. Otherwise, the calibration methods mentioned would not be possible to carry out.

4.2 T5

T5 is unified transfer learning model that uses transformers to perform text-to-text mappings (Raffel et al., 2019). The "unified" part of the name refers to the consideration of all NLP tasks as a text-to-text mapping ask, while the "tranfer learning" part of the name refers to the enhancement of various forms of pre-training before accomplishing downstream tasks.

For the T5 experiments, we used the fact that as part of its pre-training the model trained on sentiment classification tasks, specifically for SST-2. Then, it is possible to configure the model's output and perform instruction tuning and few shot learning on the inputs so that the model such that it ouputs a label for the sentiment analysis (Chung et al., 2022). We tokenize and transform the IMDB test set such that T5 can be run on it and receive the model outputs before converting to characters and the softmax. In testing, we determined that the model would always output a label for the task setup and that the maximum of the logits for the positive and negative tokens perfectly predicted the model output. So, for computational feasibility we define the normalized logits for each of the calibration methods as the normalized vector of model outputs for each label token on the classification task. Then, we perform each of the calibration methods on the logits and the gold labels of the dataset.

5 Results

5.1 LUKE

For the LUKE model outputs, the calibration methods had varied success on minimizing ECE and improving F1 scores. Histogram Binning and Isotonic Regression were the least successful, in that they tended to increase the ECE, as opposed to minimizing it. Matrix and Temperature Scaling, however, had the desired effect of minimizing ECE by several orders of magnitude.

Histogram Binning and Isotonic Regression likely had limited success due to the limitation of their being binary classification calibrators applied to a multi-class problem. We see this being the most likely factor to have limited its utility. Matrix Scaling and Temperature Scaling both did very well in terms of minimizing ECE.

Of all calibration methods, only Matrix Scaling led to a lower F1 score than the uncalibrated outputs. Overall, the differences were minor between each of the calibration methods, but still meaningfully higher than the previous output.

5.2 T5

Unike for the LUKE model outputs, Histogram Binning and Isotonic Regression did decrease ECE and improve F1 scores with respect to the uncalibrated scores. This is likely because T5 was evaluated on a binary classification task as opposed to the multi-class classificaiton task for LUKE.

Temperature Scaling and Matrix Scaling were still the better calibration methods, though. This is likely because they have more expressibility in how they can transform the data. Moreover, dividing by the temperature did not change the F1-score compared to the uncalibrated predictions.

Overall, the experiments for T5 supported the selection of the best calibration methods but implied that the decision to focus on the logits of the token of the label was the incorrect approach due to bizarre numbers. Moreover, in some splits of the data to the isotonic regression module the model would perfectly fit the data and achieve zero ECE and 1.0 F1-score.

5.3 Tables and figures [1][2][3][4][5]

The figures outputted from our experiments are shown below:

Model	UC	HB	IR	MS	TS
LUKE	.495	.586	.587	0.0055	.0122
T5	.271	.230	.201	.012	.088

Table 1: Table of output ECE using models paired with calibration methods. Note: The aim is to minimize ECE.

Model	UC	HB	IR	MS	TS
LUKE	.942	.944	.945	.939	.943
T5	.023	.044	.149	.270	.023

Table 2: Table of output F1 scores using models paired with calibration methods. Note: The aim is to maximize F1 scores.

6 Conclusion

LLMs continue to achieve state of the art performance on many benchmarks and are useful for many different tasks. But, they continue the trend of models with more layers and training data being less calibrated. Happily, Matrix Scaling and temperature scaling can well-calibrate the models.

Limitations

Within the LUKE calibration experiment, when selecting the logits to be used for calibration, We elected to use the very same logits that the model greedily selects for each word in post-processing, discarding other logits as described in the "experiments" section above. To an extent, this makes calibration slightly less effective seeing as we cannot operate on all available logits - that being said, a compromise must be made here in order to match logits to gold labels in a one-to-one fashion. There is no clear method to acquire gold labels for each and every possible span of words. Otherwise, the calibration methods mentioned would not be possible to carry out.

For the T5 calibration experiments, it is at least unprincipled that the logits for the calibration methods can be isolated to the logits for each of the tokens or sequence of tokens used as the labels for the classification task. More experimentation and evaluation is necessary to determine the validity of the taken approach. If invalid, that may explain some of the more perplexing results. Moreover, the experiments should be extended to the multi-class classification case. More experiments could determine why Isotonic Regression achieves zero ECE and 1.0 F1-score on some data splits.

[1] UC = Uncalibrated
[2] HB = Histogram Binning
[3] IR = Isotonic Regression
[4] MS = Matrix Scaling
[5] TS = Temperature Scaling

References

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep residual learning for image recognition.

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. 2018. Densely connected convolutional networks.

Hiroyuki Shindo Ikuya Yamada, Akari Asai, Hideaki Takeda, and Yuji Matsumoto. 2020. Luke: Deep contextualized entity representations with entity-aware self-attention.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. 2023. A comprehensive overview of large language models.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition.