Classifier Probability Calibration Through Uncertain Information Revision

Sara Kebir, Karim Tabia

Classifier probability calibration through uncertain information revision

(Preprint version)

Sara Kebir[1] [0000−0002−4471−9119] and Karim Tabia[1] [0000−0002−8632−3980]

Univ. Artois, CNRS, CRIL
F-62300 Lens, France
{kebir,tabia}@cril.fr

Abstract. There has been a lot of interest in explainable and trustworthy machine learning over the past few years. In some problems, it is not enough to predict correctly the true class label, but also to provide the probability that the prediction is true. This probability makes it possible to have confidence or not in the prediction. In this paper, we propose a new approach to calibrate the probabilities of a machine learning model through a post-processing step. The objective is to exploit some positive results such as that calibration is rather better on a small number of categories or subsets of classes than on a large number of classes. Based on this observation, our calibration approach, based on probabilistic belief revision, calibrates the predicted probabilities on the classes with the probabilities of subsets of classes. Preliminary experimental studies show very promising results, especially on certain datasets such as those of hierarchical classification.

Keywords: Classification · Probability calibration · Belief revision

1 Introduction

In AI, the current decade has seen the increasing use of innovative intelligent systems and applications that rely heavily on machine learning (ML). We are then confronted with new and difficult problems and risks induced, for the most part, by the complexity of the systems, their opacity and the sensitivity of certain critical applications. Among the serious questions we find confidence, trust, interpretability and explainability of reasoning and decisions that can be made automatically. Therefore, many issues are currently challenging the ML community to strengthen the explainability, interpretability and reliability of ML models and AI-based systems more generally.

In classification tasks, the goal is typically to learn a model that predicts the class variable as accurately as possible. Sometimes this is not enough since not only do we need to predict the class with good accuracy, but also provide a probability that the prediction is correct. This probability makes it possible to know the confidence that the model has in its prediction, and this can have consequences on how this prediction is managed in the context of automated decision-making as in safety-critical applications. Some models, like Bayesian classifiers, can provide directly posterior probabilities while other classifiers use certain techniques and tricks to provide these probabilities. In practice, many models give poor estimates of predictive probabilities [13], often

they overestimate these probabilities as in the case of random forests and even modern deep network-based models [5]. Calibration techniques are often used to better calibrate the probabilities of the model when it makes predictions.

We propose in this paper an original idea never used to calibrate the probabilities of a classifier. We see calibration as an uncertain information update task in the light of new uncertain and more reliable inputs. We place ourselves within the framework of Jeffrey's revision [12], a well-known framework for updating probabilistic beliefs with uncertain inputs. The preliminary results obtained confirm this intuition, in particular on certain datasets where there are class taxonomies, as is the case in several fields.

This paper is organized as follows : The second section summarizes some basic background notions about classification, probability calibration of classifiers and updating uncertain information with new uncertain inputs. Section 3 presents our work's main idea, followed in Section 4 by details about our proposed Jeffrey's rule-based probability calibration approach. Section 5, presents the experimental results, and Section 6 concludes this paper with some concluding remarks.

2 Basic background notions

2.1 Classification

Classification is a predictive task consisting in associating input data instances with symbolic labels. A classification task is defined by two sets of variables: A set of features X= $\{X_1,..,X_n\}$ where $|X|$=n, and a discrete target variable denoted C taking values in its domain D_C.

Definition 1. ***(Classifier)*** *A classifier f is a function mapping each input data instance x (vector instantiating each variable in X) to one value from the discrete variable domain D_C.*

Note that classification where each data instance x is associated exactly with one class is called multi-class classification, contrary to multi-label classification where one can associate a subset of classes at the same time to a data instance. In this paper, we deal only with calibrating multi-class classifiers.

2.2 Classifier probability calibration

We say that a classifier f is calibrated (or provides calibrated prediction probabilities) if, when it predicts a label $c_i \in C$ with probability $\hat{\mathrm{p}}_i$, this prediction will be correct with probability $\hat{\mathrm{p}}_i$ (intuitively, the probability $p_f(C{=}c_i|p{=}\hat{p}_i)$ is calibrated if on average, the prediction is correct with probability $\hat{\mathrm{p}}_i$).

In order to visualize the quality of predicted probabilities, one can display a *reliability diagram* (see example of Fig.1), a visual representation of prediction calibration. This comes down to plotting expected accuracy as a function of confidence. In such a diagram, confidence estimates are grouped into bins to allow computing the sample accuracy. Hence, a well calibrated model corresponds to the plot of the identity function. Calibration errors correspond to the gap between the estimated probabilities and the accuracy ones.

As for measuring classifiers efficiency, there are different measures for measuring miscalibration. Some commonly used metrics are Expected Calibration Error (ECE), Average Calibration Error (ACE), and Maximum Calibration Error (MCE). In few words, miscalibration metrics assess the errors by binning the samples by their confidence then assess the accuracy in each bin. For instance, ECE is simply the weighted mean gap between the confidence of the classifier and the observed accuracy (on a test set) in each bin. Similarly, the Maximum Calibration Error (MCE) gives simply the maximal gap (see for instance [8] for more details on measuring miscalibration). Negative log likelihood (NLL) can also be used to indirectly measure the model calibration since it penalizes high probability scores assigned to incorrect labels and low probability ones assigned to correct labels [1]. The lower these metrics are, the better is the quality of the calibration.

Various approaches have been proposed to perform recalibration. Platt scaling [11] and isotonic regression [9] are the most widely used in the binary classification setting. The most common way of extending these methods to the multi-class setting is treating the problem as k one-versus-all problems, where k is the number of classes [15]. Below, we recall how the two main post-processing methods perform.

Platt scaling is a parametric calibration method which fits a logistic regression model on the validation set, using the non-probabilistic predictions of the initial classifier z_i as features, to learn scalar parameters $a, b \in \mathbb{R}$ and compute the calibrated probabilities $\hat{p}_i$ as $\hat{p}_i = \alpha(az_i + b)$. The parameters a and b can be optimized by minimizing the Negative Log Likelihood. The Platt scaling can be extended to the multi-class setting by applying a linear transformation $Wz_i + b$ to the logits vector z_i. The parameters a and b will be in a higher dimension, $W \in \mathbb{R}^{k*k}$ and $b \in \mathbb{R}^k$ respectively. The resulting method is called **Matrix scaling** or **Vector scaling** if W is a diagonal matrix. There is also another extension of this method, called **Temperature scaling**, which uses a single scalar parameter $T > 0$ to calibrate the probabilistic predictions as follows $\hat{p}_i = \max_l \alpha_{sm}(z_i/T)^{(l)}$, where $\alpha_{sm}(z_i)^{(l)}$ is the predicted probability of the class $l = 1..k$ [5].

Isotonic regression is a non-parametric calibration method which outputs a piecewise constant function f to transform a probability p_i into a calibrated one by minimizing the mean-squared loss function $\sum_{i=1}^{n}(f(p_i) - y_i)^2$, where y_i is the true label. The Isotonic regression can be extended to the multi-class setting by performing a calibration for each class separately as a one-versus-all problem followed by a postprocessing to normalize the combined calibrated probabilities.

2.3 Updating uncertain information with new uncertain inputs : Jeffrey's rule

Jeffrey's rule [6] extends the classical probabilistic conditioning to the case where the new information is uncertain. It allows to update an initial probability distribution p into a posterior one p' given the uncertainty bearing on a set of mutually exclusive and exhaustive events λ=$\{\lambda_1,..,\lambda_n\}$ (namely, λ is a partition of the set of possible states Ω). In this setting, the new input is in the form (λ_i, α_i), i=1..n where α_i denotes the new probability of λ_i. Jeffrey's rule lies on the two following principles:

- Success principle (P1)

$$\forall \lambda_i \in \lambda, p'(\lambda_i) = \alpha_i \tag{1}$$

After the update operation, the posterior probability of each event λ_i must be equal to α_i as required in the new inputs. The uncertain inputs are seen as constraints or an effect once the new information is fully accepted.

- Probability kinematics principle (P2)

$$\forall \lambda_i \in \lambda, \forall \phi \subseteq \Omega, p(\phi|\lambda_i) = p'(\phi|\lambda_i) \tag{2}$$

This principle aims to ensure a kind of minimal change by ensuring that the posterior distribution p' should not change the conditional probability degrees of any event ϕ given the uncertain events λ_i. Jeffrey's rule assumes that in spite of the disagreement about the events λ_i in the prior distribution p and the posterior one p', the conditional probability of any event $\phi \subseteq \Omega$ given any uncertain event λ_i should remain the same in the original and the revised distributions.

Given a probability distribution p encoding the initial beliefs and new inputs in the form (λ_i, α_i) for i=1..n, the updated probability degree of any event $\phi \subseteq \Omega$ is obtained as follows:

$$p'(\phi) = \sum_{\lambda_i} \alpha_i * \frac{p(\phi \cap \lambda_i)}{p(\lambda_i)} \tag{3}$$

The posterior distribution p' obtained using Jeffrey's rule always exists and it is unique [4]. Note that in Jeffrey's rule, the events λ_i should be somewhat possible in the prior distribution (namely, $\forall \lambda_i \in \lambda$, $p(\lambda_i) > 0$).

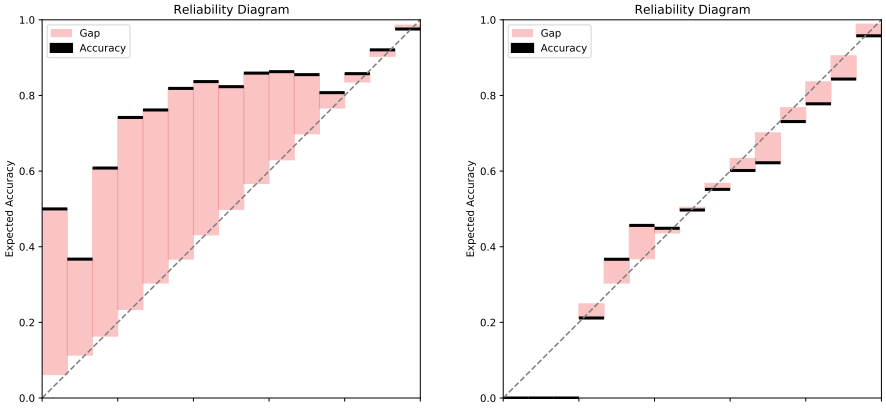
Jeffrey's framework for updating uncertain information with uncertain inputs has been used primarily for reasoning with uncertain information and observations [10]. In classification, it has been used instead to make predictions with uncertain observations (often called soft evidence) in some classifiers [2,3] such as those based on Bayesian networks or recently with some neural networks-based classifiers [14]. To the best of our knowledge, none of these works had used Jeffrey's rule of conditioning for probability calibration purposes.

3 Motivating example

The starting point of this work is the following observation: on large-scale classification problems (involving a large number of classes), it is often difficult to correctly predict certain classes, especially in the case of unbalanced datasets (often, classifiers favor majority classes to the detriment of underrepresented classes). What is true for the accuracy of the predictions is also true for the confidence probabilities of the model in its predictions. This difficulty may be essentially linked to the nature of the data and the specificities of the classifiers used. For example, if we use random forests by limiting the depth of the trees too much, it will be difficult to discriminate certain classes since the number of tests on the features is strongly limited.

Two simple questions then arise: i) If we group classes into categories (kind of super classes, so as to be better represented and reduce the number of classes), can we improve the quality of the predictions (in terms of accuracy and calibration)? ii) If so, would it be possible to exploit the best performances of the predictions on the categories (subsets of the initial classes) to *rectify* or *calibrate* the predictions and the calibration of the classifier f?

For the first question, the answer is positive for most of the datasets and classifiers, although with different results depending on how the classes $\{c_1, .., c_k\}$ are grouped into categories $\{cat_1, .., cat_j\}$ (with $j<k$). In Fig. 1, one can see the reliability diagrams of an SVM classifier learnt on the well-known DBPedia[1] dataset. Clearly, the SVM classifier built on 70 classes is poorly calibrated (see the gap to the perfect calibration line) compared to the SVM built on 9 categories.



(a) SVM 70 (b) SVM 9

Fig. 1: Reliability diagrams of SVM classifiers on DBPedia

For the second question, having a more calibrated classifier on the categories provides relevant information on the classes included in each category. How to use these probabilities on the categories and in which case they should be used in order to guarantee an improvement of the probabilities of the initial classifier will be discussed in the following section.

4 Probability calibration based on Jeffrey's rule

Let us denote the classifier to calibrate f and let us denote by f' the one learnt on categories and ensuring better calibrated probabilities on categories. Each category from

[1] This dataset is a cleaned extract of 342,782 wikipedia articles' data providing hierarchical classes (there are 3 levels with 9, 70 and 219 classes respectively), https://www.kaggle.com/danofer/dbpedia-classes.

$\{cat_1, .., cat_j\}$ is a subset of classes from $\{c_1, .., c_k\}$ (note that a given class belongs to only one category). Namely, the categories form a partition of the set of classes.

4.1 Jeffrey's rule-based Probability Calibration

We have, on one side, classes and a probability distribution provided by the classifier f, and on the other side, categories (a partition of the classes) and another probability distribution on categories provided by the classifier f'. Moreover, in most cases, the probabilities of the classifiers f' are more calibrated as illustrated in the example of Fig. 1 meaning that the classifier f' is more reliable in terms of calibration. This places our problem somehow in the framework of updating uncertain information with new uncertain information. Note that the prior information is a probability distribution p over $\{c_1, .., c_k\}$ and that the new information is also uncertain and it is in the form of a probability distribution p' over a partition of $\{c_1, .., c_k\}$ and denoted $\{cat_1, .., cat_j\}$. It fully makes sense to update the distribution p with p' since this latter provides more calibrated probabilities. This comes down to give priority to the new information p' exactly in line with Jeffrey's rule. Hence, the revised probabilities p_c are obtained following Jeffrey's rule as follows : $\forall c_i \in D_C$,

$$p_c(c_i) = p(c_i) * \frac{p'(cat(c_i))}{p(cat(c_i))}, \tag{4}$$

where $cat(c_i)$ denotes the category of class c_i. Note also that $p(cat(c_i))$ is the probability of all classes from category $cat(c_i)$ computed from the prior distribution p. The posterior distribution p_c always exists and it is unique unless the first classifier f associates a zero probability to $cat(c_i)$ (namely, $p(cat(c_i)) = 0$).

4.2 Jeffrey's rule-based Probability Calibration in practice

Up to now, we have briefly presented the main idea to calibrate the probabilities of a classifier f by exploiting the probabilities of another classifier f' on categories in the spirit of Jeffrey's rule. Now, several questions arise regarding the use of our calibration technique in practice :

- *How to group classes into categories ?* In some domains, there are taxonomies and class hierarchies to semantically group classes into categories. This is a first option but it does not necessarily guarantee the best results. The number of categories and the composition of the categories is one of the key points to have well calibrated probabilities on the categories to ensure better results after the calibration. For datasets without class taxonomies, one way would be to cluster the data and obtain the categories corresponding to the clusters using some clustering techniques.
- *In what case there could be an improvement and how much will the improvement be?* The improvement will be all the greater when the initial probabilities are weakly calibrated and when the probabilities on the categories p' are such that they can correct the initial probabilities p. This needs to be formally or empirically characterized to identify the situations where calibration based on the proposed method

is advisable. Another idea to improve the results of the calibration is, as we will see in our preliminary experimental study, to start with calibrated probabilities both for the classifier to be calibrated f and for the calibration classifier f'. Indeed, nothing prevents in this case from using existing calibration techniques to pre-calibrate p and p' to finally revise according to our Jeffrey's rule-based Probability Calibration method. We don't even have to use the same classification technique for f and f' in case the latter guarantees a better calibration on the categories.

These questions are not trivial and are beyond the scope of this paper.

Before proceeding to the experimental study section, Fig. 2 below illustrates our Jeffrey's rule-based Probability Calibration process that encompasses both of the above issues, i.e. the part regarding how to group classes into categories and the one on how to further improve the initial calibration using an oracle (for instance, using existing calibration techniques).

5 Experimental study

This section presents preliminary results evaluating our approach for classifier probability calibration based on Jeffrey's rule. The experiments are carried out on some well-known datasets where grouping classes into categories is done thanks to existing class taxonomies (for instance, DBPedia and Amazon products reviews datasets) or where it is easy to group manually classes into categories according to the semantics of classes (such as, Stanford Sentiment Treebank dataset) or using a clustering method (as is the case for MNIST and Fashion-MNIST datasets) which are described below :

- *DBPedia dataset*[2] is a cleaned extract of 342,782 wikipedia articles' data. Widely used as a baseline for NLP/text classification tasks, it provides hierarchical categories in three levels with 9, 70 and 219 classes, respectively.
- *Amazon products reviews (Amazon PR)* [3] is a dataset of amazon customers reviews structured by products into 3 levels with 6, 64 and 510 classes, respectively.
- *Stanford Sentiment Treebank (SST)* [4] is a dataset containing 215,154 phrases with fine-grained sentiment labels in the parse trees of 11,855 sentences in movie reviews rated from 1 to 5.
- *MNIST* [7] is a handwritten digits dataset of 28x28 images containing a training set of 60,000 examples and a test set of 10,000 examples.
- *Fashion-MNIST* [5] is a dataset of Zalando's article images consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image, associated with a label from 10 classes.

We evaluate our proposed approach on four basic classifiers, namely, Naive Bayes (NB), Random Forests (RF), Logistic Regression (LR) and Support Vector Machines

[2] DBPedia dataset, https://www.kaggle.com/danofer/dbpedia-classes
[3] Amazon PR dataset, https://www.kaggle.com/kashnitsky/hierarchical-text-classification
[4] SST dataset, https://nlp.stanford.edu/sentiment/treebank.html
[5] Fashion-MNIST dataset, https://github.com/zalandoresearch/fashion-mnist

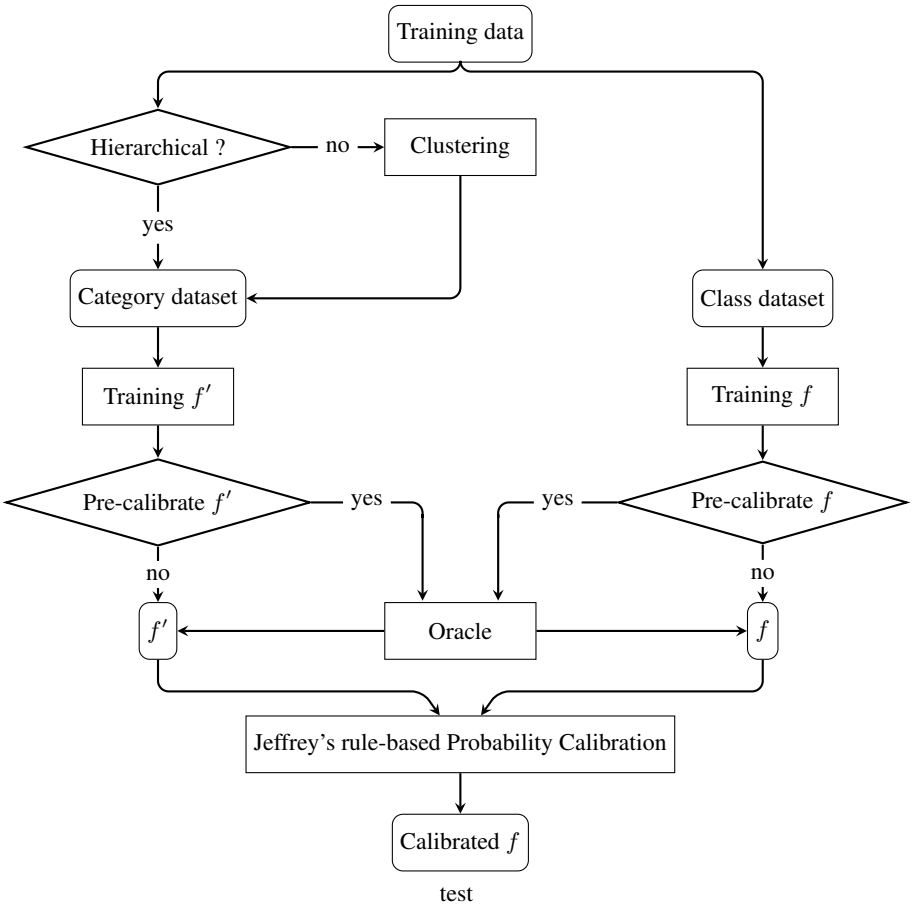


Fig. 2: Jeffrey's rule-based Probability Calibration

(SVM). These techniques have been implemented and parameterized using the scikit-learn library's basic classifiers and configuration. The evaluation metrics used to assess how accurate and how well the confidence in the model's prediction is calibrated are: Accuracy (Acc), Negative Log Likelihood (NLL), Expected Calibration Error (ECE) using 15 bins, and Maximum Calibration Error (MCE). Except for the accuracy, the lower these measures are, the better the calibration's quality will be.

The results of our probability calibration approach on Amazon product reviews and DBPedia datasets are shown in Table 1. For both datasets, we have used only two levels: level one for category labels (6, 9 categories, respectively) and level two for fine-grained class labels (64, 70 classes, respectively). As expected, except for Random Forest on Amazon PR, all tested models show an improvement in confidence quality based on all calibration evaluation metrics, while maintaining, or improving in some cases, the initial accuracy, thus confirming the potential effectiveness of our approach. Indeed, unlike the other cases, one can notice that the Random Forest was already well calibrated at the

beginning, with an ECE $\approx 5\%$, which explains the ineffectiveness of the calibration in this situation and confirms that when the initial classifier is already well calibrated, neither the state-of-the-art methods nor ours have much room for improvement, as shown in Table 2.

Table 1: Comparison between the performance of the classifiers before (Uncalibrated) and after the use of Jeffrey's rule-based Probability Calibration (Proposed method) on Amazon Products Reviews and DBPedia datasets.

Model		Amazon PR				DBPedia			
		Acc%	NLL	ECE%	MCE%	Acc%	NLL	ECE%	MCE%
NB	Uncalibrated	42.93	2.67	27.63	67.17	71.42	1.22	22.41	34.08
	Proposed method	**53.01**	**2.05**	**26.49**	**53.33**	**71.71**	**1.14**	**20.51**	**32.09**
LR	Uncalibrated	64.00	1.77	23.73	44.90	**92.30**	0.36	10.31	36.25
	Proposed method	**67.44**	**1.42**	**18.25**	**32.79**	91.76	**0.35**	**09.23**	**29.74**
RF	Uncalibrated	67.19	2.50	**05.16**	**10.59**	**90.57**	0.62	26.39	47.24
	Proposed method	**67.39**	**2.40**	14.76	25.32	90.36	**0.60**	**25.66**	**43.03**
SVM	Uncalibrated	62.40	1.65	16.85	40.86	**83.77**	**0.87**	21.16	54.12
	Proposed method	**63.35**	**1.59**	**06.78**	**14.98**	81.01	0.90	**11.05**	**30.08**

Table 2: Comparison between the performance of the classifiers before (Uncalibrated) and after the use of state-of-the-art calibration methods and Jeffrey's rule-based Probability Calibration using the oracle (Proposed method-oracle), on Amazon Products Reviews and DBPedia datasets.

Model		Amazon PR				DBPedia			
		Acc%	NLL	ECE%	MCE%	Acc%	NLL	ECE%	MCE%
NB	Uncalibrated	42.93	2.67	27.63	67.17	71.42	1.22	22.41	34.08
	Isotonic reg	68.19	1.50	13.03	24.46	**88.88**	**0.45**	12.13	**24.8**
	Sigmoid reg	62.32	1.71	22.49	28.37	83.03	0.76	16.62	25.66
	Proposed method-oracle	**70.12**	**1.38**	**9.40**	**20.67**	82.00	0.69	**8.53**	26.41
LR	Uncalibrated	64.00	1.77	23.73	44.90	**92.30**	**0.36**	10.31	36.25
	Isotonic reg	67.32	1.47	11.51	19.33	91.76	0.41	15.99	33.26
	Sigmoid reg	68.05	1.31	13.45	20.50	92.12	0.40	16.35	32.35
	Proposed method-oracle	**70.27**	**1.13**	**9.10**	**18.76**	91.50	**0.36**	**8.38**	**24.76**
RF	Uncalibrated	**67.19**	2.50	**05.16**	10.59	90.57	0.62	26.39	47.24
	Isotonic reg	64.39	5.16	21.14	46.88	**91.99**	0.39	2.71	13.30
	Sigmoid reg	67.02	**1.73**	20.79	41.86	91.47	**0.32**	1.94	11.72
	Proposed method-oracle	66.24	2.63	5.65	**9.12**	90.10	0.49	**1.41**	**8.46**
SVM	Uncalibrated	62.40	1.65	16.85	40.86	83.77	0.87	21.16	54.12
	Isotonic reg	**65.89**	**1.52**	11.03	22.16	**85.49**	**0.64**	18.35	30.88
	Sigmoid reg	24.59	2.65	**6.12**	33.11	33.67	2.55	**8.76**	40.95
	Proposed method-oracle	63.35	1.59	6.78	**14.98**	81.01	0.90	11.06	**30.08**

To further improve the calibration performance, we apply our approach on the resulting models from the oracle which have, in the most cases, a pre-calibrated probabilities p and $p^{'}$. The pre-calibration is provided through the use of state-of-the-art calibration techniques or another classifier for f', trained on the same training set and showing better calibrated probabilities $p^{'}$ on the categories. The obtained results are shown in Tables 2, 3 and 4.

As expected, the previous results, illustrated in Table 1, improve further with the use of an oracle and outperform state-of-the-art models, namely, Isotonic regression and Sigmoid regression. We can see for instance in Table 2, the Random Forest on DBPedia related ECE being reduced from 26.39 to 1.41%, while ensuring a good accuracy. The same applies to almost all other classifiers on those datasets. In most cases, the oracle result is a pre-calibration of the probabilities p and $p^{'}$ using Isotonic regression since when comparing the results of the two state-of-the-art calibration methods, the latter performs better.

The Stanford Sentiment Treebank (SST) dataset has been processed differently. As it does not have any levels, a manual clustering based on the semantic of its classes, films rating, is used to group them into 2 ([[1,2],[3,4,5]]) and 3 ([[1,2],[3],[4,5]]) clusters. The results illustrated in Table 3 confirm the effectiveness of our calibration approach on this dataset too. In addition to the calibration effect, one may notice that the initial evaluation metrics are much lower compared to the results presented previously.

Table 3: Comparison between the performance of the classifiers before (Uncalibrated) and after the use of state-of-the-art calibration methods and Jeffrey's rule-based Probability Calibration using the oracle (Proposed method-oracle), on SST.

	Model	SST			
		Acc%	NLL	ECE%	MCE%
NB	Uncalibrated	36.20	1.49	3.26	5.93
	Isotonic reg	39.77	1.40	3.76	6.53
	Sigmoid reg	**40.14**	1.42	5.79	9.59
	Proposed method-oracle	39.14	**1.38**	**2.80**	**5.67**
LR	Uncalibrated	36.47	1.43	2.37	**7.32**
	Isotonic reg	39.55	1.39	2.95	26.54
	Sigmoid reg	38.46	1.41	2.89	13.33
	Proposed method-oracle	**39.82**	**1.37**	**0.91**	8.12
RF	Uncalibrated	33.94	2.51	20.81	45.87
	Isotonic reg	37.10	1.45	2.83	4.78
	Sigmoid reg	**37.38**	1.47	4.27	**7.20**
	Proposed method-oracle	36.38	**1.44**	**1.48**	21.50
SVM	Uncalibrated	36.83	1.52	12.71	49.87
	Isotonic reg	38.55	1.40	1.71	26.38
	Sigmoid reg	36.61	1.42	**1.43**	**18.75**
	Proposed method-oracle	**39.77**	**1.38**	2.33	32.25

To get the required categories from datasets without class taxonomies, as is the case for MNIST and Fashion-MNIST, and where a semantic categorisation as separating the handwritten digits of MNIST dataset into odd and even numbers proved to be ineffective, a k-means clustering technique is performed to group the classes into 2 and 3 clusters. The results obtained with the application of our proposed calibration approach on this different type of datasets are given in Table 4 and confirm, once again, the effectiveness of our proposed Jeffrey's rule-based Probability Calibration technique.

Table 4: Comparison between the performance of the classifiers before (Uncalibrated) and after the use of state-of-the-art calibration methods and Jeffrey's rule-based Probability Calibration using the oracle (Proposed method-oracle), on MNIST and Fashion-MNIST datasets.

Model		MNIST				Fashion-MNIST			
		Acc%	NLL	ECE%	MCE%	Acc%	NLL	ECE%	MCE%
NB	Uncalibrated	83.57	1.99	13.93	45.26	65.52	5.55	29.52	59.56
	Isotonic reg	84.69	0.50	1.37	7.17	70.17	0.86	3.46	11.76
	Sigmoid reg	83.65	0.73	4.63	38.45	65.68	1.24	8.45	25.90
	Proposed method-oracle	**88.54**	**0.36**	**0.76**	**5.67**	**72.54**	**0.73**	**2.64**	**10.63**
LR	Uncalibrated	92.56	0.27	0.68	**6.98**	84.43	0.44	2.16	7.26
	Isotonic reg	90.95	0.46	17.49	29.74	77.12	0.80	21.18	32.90
	Sigmoid reg	91.33	0.47	20.14	33.76	78.89	0.83	26.42	36.43
	Proposed method-oracle	**94.88**	**0.18**	**0.64**	13.68	**85.43**	**0.41**	**0.91**	**5.90**
RF	Uncalibrated	**97.05**	0.24	13.65	38.27	87.72	**0.41**	7.86	**18.35**
	Isotonic reg	95.62	0.61	0.23	**11.44**	86.16	2.95	0.85	39.85
	Sigmoid reg	97.00	**0.12**	1.75	20.17	**87.73**	0.64	9.53	40.45
	Proposed method-oracle	95.60	0.85	**0.16**	15.44	86.21	2.94	**0.58**	40.38
SVM	Uncalibrated	95.85	0.13	1.55	11.66	**86.87**	**0.37**	0.76	5.49
	Isotonic reg	95.77	0.21	**0.47**	35.37	86.80	0.47	2.72	20.84
	Sigmoid reg	95.78	0.32	2.46	36.38	86.81	0.56	2.12	11.67
	Proposed method-oracle	**97.24**	**0.10**	0.95	**9.96**	86.77	**0.37**	**0.56**	**3.05**

6 Concluding remarks

In this preliminary work, we have sketched out a novel method to calibrate the probabilities of a classifier through uncertain and more reliable information revision based on Jeffrey's rule of conditioning.

We have noticed during the experimental study that the NLL, ECE and MCE metrics are not sufficient to predict the effect of the probability calibration technique. Using a category model that is either overconfident or underconfident while displaying the same calibration measures in both cases does not lead to the same results after the revision, in other words, trying to calibrate a very overconfident classes model with another overconfident category model is not very useful, the same applies to underconfidence.

One of the most important issues facing the proposed approach is the quality of the category model. Thus, future work may focus on finding a better way of categorisation

since the one obtained with the clustering method or the given class taxonomies in hierarchical datasets are not necessarily good.

The computational complexity of our calibration approach depends on the one of calculating the two distributions p and p'. There are two distinct cases. If we place ourselves in the case without using an oracle, we will have to sum the cost of calling the category classifier to get p' and the one of applying our calibration method, which is linear in the number of classes, whereas in the other case we will have to add the cost of the oracle calls.

Even though we have only employed the most basic machine learning classifiers so far, the calibration approach proposed in this paper is quite competitive with state-of-the-art methods and has demonstrated its efficacy. We are aware that a wide range of highly accurate and complex classification models exist, and that highly efficient calibration techniques have already been proposed and applied to them; therefore, our next step is to test our proposed calibration approach on them, as well as to try to expand the range of datasets used to further provide evidence on the effectiveness of our approach.

Acknowledgement. This work has been supported by the Vivah project 'Vers une Intelligence artificielle à VisAge Humain' supported by the ANR.
This work has received support from the ANR CROQUIS (Collecting, Representing, cOmpleting, merging, and Querying heterogeneous and UncertaIn waStewater and stormwater network data) project, grant ANR-21-CE23-0004 of the French research funding agency Agence Nationale de la Recherche (ANR).

References

1. Ashukha, A., Lyzhov, A., Molchanov, D., Vetrov, D.P.: Pitfalls of in-domain uncertainty estimation and ensembling in deep learning. In: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net (2020), https://openreview.net/forum?id=BJxI5gHKDr
2. Benferhat, S., Tabia, K.: Inference in possibilistic network classifiers under uncertain observations. Ann. Math. Artif. Intell. **64**(2-3), 269–309 (2012). https://doi.org/10.1007/s10472-012-9290-1, https://doi.org/10.1007/s10472-012-9290-1
3. Benferhat, S., Tabia, K.: Reasoning with uncertain inputs in possibilistic networks. In: Principles of Knowledge Representation and Reasoning: Proceedings of the Fourteenth International Conference, KR 2014, Vienna, Austria, July 20-24, 2014 (2014), http://www.aaai.org/ocs/index.php/KR/KR14/paper/view/7964
4. Chan, H., Darwiche, A.: On the revision of probabilistic beliefs using uncertain evidence. Artif. Intell. **163**(1), 67–90 (2005)
5. Guo, C., Pleiss, G., Sun, Y., Weinberger, K.Q.: On calibration of modern neural networks. In: Precup, D., Teh, Y.W. (eds.) Proceedings of the 34th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 70, pp. 1321–1330. PMLR (06–11 Aug 2017), https://proceedings.mlr.press/v70/guo17a.html
6. Jeffrey, R.: The Logic of Decision. McGraw Hill, New York (1965)
7. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998). https://doi.org/10.1109/5.726791

8. Naeini, M.P., Cooper, G.F., Hauskrecht, M.: Obtaining well calibrated probabilities using bayesian binning. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence. p. 2901–2907. AAAI'15, AAAI Press (2015)
9. Niculescu-Mizil, A., Caruana, R.: Predicting good probabilities with supervised learning. In: Proceedings of the 22nd International Conference on Machine Learning. p. 625–632. ICML '05, Association for Computing Machinery, New York, NY, USA (2005). https://doi.org/10.1145/1102351.1102430, https://doi.org/10.1145/1102351.1102430
10. Peng, Y., Zhang, S., Pan, R.: Bayesian network reasoning with uncertain evidences. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems **18**, 539–564 (10 2010). https://doi.org/10.1142/S0218488510006696
11. Platt, J.: Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. Advances in large margin classifiers **10**(3), 61–74 (1999)
12. Shafer, G.: Jeffrey's rule of conditioning. Philosophy of Science **48**(3), 337–362 (1981), http://www.jstor.org/stable/186984
13. de Menezes e Silva Filho, T., Song, H., Perelló-Nieto, M., Santos-Rodríguez, R., Kull, M., Flach, P.A.: Classifier calibration: How to assess and improve predicted class probabilities: a survey. CoRR **abs/2112.10327** (2021), https://arxiv.org/abs/2112.10327
14. Yu, E.: Bayesian neural networks with soft evidence. CoRR **abs/2010.09570** (2020), https://arxiv.org/abs/2010.09570
15. Zadrozny, B., Elkan, C.: Transforming classifier scores into accurate multiclass probability estimates. In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. p. 694–699. KDD '02, Association for Computing Machinery, New York, NY, USA (2002). https://doi.org/10.1145/775047.775151, https://doi.org/10.1145/775047.775151