Bayesian Estimation of Prediction Uncertainty in Speaker Identification

Jan Zdeněk

October 30, 2025

Abstract

This work applies Monte Carlo Dropout to an embedding-based speaker identification system to estimate predictive uncertainty. While deterministic models achieve slightly higher accuracy, Monte Carlo Dropout provides meaningful confidence measures such as predictive entropy and variation ratio. The results demonstrate its ability to capture uncertainty and improve reliability under ambiguous or noisy speech conditions.

1 Introduction

Speaker Identification (SID) systems aim to predict the identity of a speaker from an audio signal. Modern such systems leverage deep embeddings extracted from neural networks, such as x-vectors (Snyder et al., 2018) or ECAPA-TDNN (Desplanques et al., 2020). These embeddings are then fed into a classifier head trained to distinguish between already known speakers. Even though such deterministic models usually achieve high accuracy, they provide no information about the prediction confidence. This lack of confidence can pose a problem, especially in security or forensic applications.

Deterministic neural networks produce only point predictions under the assumption of complete parameter certainty, which rarely holds in practice. In situations where the signal is noisy or ambiguous, they produce overly confident and potentially misleading outputs (Gal and Ghahramani, 2016; Kendall and Gal, 2017). Reliable systems, therefore, require a mechanism to quantify predictive uncertainty, enabling a distinction between a confident and ambiguous decision.

The Monte Carlo (MC) Dropout method provides a practical approximation to Bayesian inference in deep learning (Gal and Ghahramani, 2016). By keeping dropout layers active even at inference time and averaging outputs from multiple stochastic forward passes, the model captures epistemic uncertainty without requiring any changes to the underlying model architecture or the training procedure.

This work implements and evaluates MC Dropout in an embedding-based SID framework. It compares deterministic and stochastic approaches, focusing on uncertainty metrics such as predictive entropy and variation ratio, and analyzes how uncertainty relates to prediction correctness across speakers. The aim is not to maximize classification accuracy but to illustrate the behavior of uncertainty estimation under challenging conditions, using a deliberately difficult dataset and a simplified model architecture.

2 Model

The proposed SID system follows an embedding-based architecture consisting of two stages: an embedding backend and a cosine-normalized classification frontend. The backend is a pretrained neural network that transforms a variable-length speech signal $x(t)$ into a fixed-dimensional embedding vector $\mathbf{e} \in \mathbb{R}^D$, where D denotes the embedding dimension. This process can be viewed as a function with signature $f_{enc} : \mathcal{X} \to \mathbb{R}^D$, such that $\mathbf{e} = f_{enc}(x)$. In this setup, the backend serves solely as a feature extractor and remains frozen during training.

The frontend classifier is implemented as a lightweight single-hidden-layer perceptron with dropout regularization and cosine normalization in the final stage. Given an embedding $\mathbf{e}$, the hidden representation is computed as $\mathbf{h} = \phi(\,\mathbf{W}_1\mathbf{e} + \mathbf{b}_1\,)$, where $\phi(\cdot) = \mathrm{ReLU}(\cdot)$ and dropout is applied both before and after the linear layer to encourage robustness and enable MC Dropout during inference.

The classifier head replaces the standard linear layer with a cosine-similarity-based approach. Let $\mathbf{W}_2 \in \mathbb{R}^{C\times H}$ denote the weight matrix of the output layer, where C is the number of speaker classes and H is the hidden dimension. Both the feature vector and the class weights are ℓ_2-normalized:

$$\hat{\mathbf{h}} = \frac{\mathbf{h}}{\|\mathbf{h}\|_2}, \qquad \hat{\mathbf{W}}_2 = \frac{\mathbf{W}_2}{\|\mathbf{W}_2\|_2}.$$

[1] The output logits are then scaled cosine similarities $\mathbf{z} = s\hat{\mathbf{W}}_2\hat{\mathbf{h}}$, where $s \in \mathbb{R}^+$ is a learnable scaling parameter that controls the magnitude. The posterior class probabilities are obtained using the softmax function:

$$p(y = c \,|\, \mathbf{e}) = \frac{\exp(z_c)}{\sum_{k=1}^{C} \exp(z_k)}.$$

This approach encourages angular separation between speakers and is particularly suitable for embedding-based recognition systems (Wang et al., 2018; Deng et al., 2022).

3 Methods

The proposed approach follows a standard supervised training pipeline, where only the classification head is optimized while the pretrained embedding network remains frozen. Each training sample is a fixed-dimensional embedding vector extracted from a speaker utterance. The model is trained using the cross-entropy loss between predicted class probabilities and ground-truth speaker labels. The optimizer is AdamW with a learning rate of 10^{-3} and a weight decay of 10^{-4}. Training is performed for a fixed number of epochs with mini-batches of size 128. Dropout regularization with rate $p = 0.3$ is applied both before and after the hidden layer to reduce overfitting and to enable MC Dropout inference.

During inference, the standard deterministic evaluation is complemented by a stochastic MC Dropout procedure. Instead of disabling dropout, the model is kept in training mode, and $T = 100$ stochastic forward passes are performed for each test sample. The mean predictive probability

$$\bar{p}(y|x) = \frac{1}{T}\sum_{t=1}^{T} p_t(y|x)$$

[1] $\|\cdot\|$ applied to a matrix denotes a row-wise ℓ_2-normalization.

is used to estimate the final prediction, while predictive uncertainty is quantified using the entropy

$$H(x) = -\sum_c \bar{p}(y = c|x) \log \bar{p}(y = c|x)$$

and the variation ratio $V(x) = 1 - \max_c \bar{p}(y = c|x)$. These metrics together capture the model's epistemic uncertainty and provide insight into the confidence of each decision.

All experiments are implemented in PyTorch using standard scientific Python libraries (NumPy, Matplotlib, scikit-learn) for data handling and visualization. The experiments are run on macOS with CPU and MPS acceleration.

4 Numerical Examples

Experiments are conducted on the CN-Celeb1, VCTK, and LibriSpeech datasets using the same model setup described in Section 2. The CN-Celeb1 corpus presents a challenging, real-world test scenario, whereas VCTK and LibriSpeech contain much cleaner and higher-quality speech. The results of both deterministic and MC Dropout evaluations are summarized in Table 1. Stochastic inference slightly decreases accuracy while providing uncertainty estimates through entropy and variation ratio.

Dataset	**Model**	**Accuracy**	**Mean Entropy**	**Mean Var. Ratio**
CN-Celeb1	Deterministic	0.7753	–	–
	MC Dropout (T = 100)	0.7602	3.8002	0.6581
VCTK	Deterministic	0.9998	–	–
	MC Dropout (T = 100)	0.9997	0.2498	0.0396
LibriSpeech	Deterministic	0.9993	–	–
	MC Dropout (T = 100)	0.9987	0.8528	0.1115

Table 1: Comparison of deterministic and MC Dropout evaluation on all datasets – CN-Celeb1, VCTK, and LibriSpeech.

All visualizations in this section are based on the CN-Celeb1 dataset. Figure 1 presents the per-class probability spread for a single sample under MC Dropout inference. In this particular example, two classes exhibit high but highly variable probabilities, revealing that the model is uncertain between them – a pattern that would remain hidden in a deterministic prediction. Figure 2 compares predictive-entropy distributions between correct and incorrect predictions, and Figure 3 shows per-speaker mean entropy together with individual accuracies.

5 Conclusions

The project successfully implemented MC Dropout within an embedding-based SID system to evaluate predictive uncertainty. All experiments were conducted using PyTorch and standard Python scientific libraries, which proved easy to install and well-suited for rapid experimentation. The results confirmed that while deterministic models achieved slightly higher accuracy, the MC Dropout approach provided uncertainty estimates that reflected model confidence. This made the system more interpretable, particularly for difficult speech samples. Future work could explore more advanced Bayesian inference methods or integrate uncertainty estimates into decision-making thresholds.

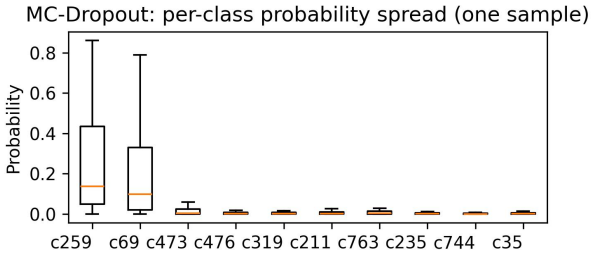

Figure 1: Boxplots show the distribution of softmax probabilities across stochastic passes for the top-K classes. This illustrates how MC Dropout provides a per-sample uncertainty that can be thresholded in practice.

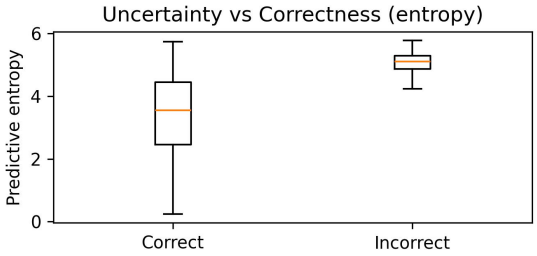

Figure 2: Predictive entropy for samples. Incorrect samples show higher and more concentrated entropy, while correct ones span a wider range, indicating that entropy reliably reflects model confidence.

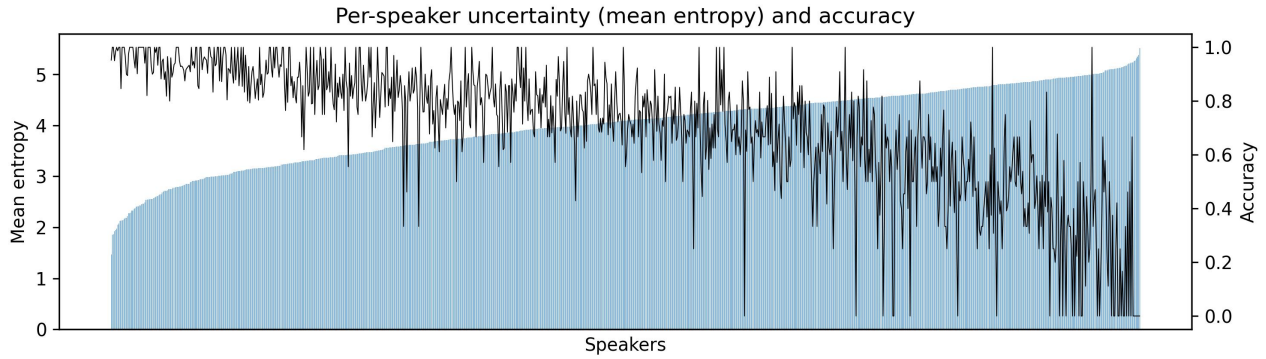

Figure 3: Per-speaker mean entropy (bars) and accuracy (line), with speakers sorted by entropy. The plot reveals an inverse relationship between uncertainty and accuracy – speakers with higher entropy tend to be harder to classify correctly.

References

Deng, J., Guo, J., Yang, J., Xue, N., Kotsia, I., and Zafeiriou, S. (2022). Arcface: Additive angular margin loss for deep face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):5962–5979.

Desplanques, B., Thienpondt, J., and Demuynck, K. (2020). Ecapa-tdnn: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification. In *Interspeech 2020*, interspeech_2020. ISCA.

Gal, Y. and Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning.

Kendall, A. and Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision?

Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., and Khudanpur, S. (2018). X-vectors: Robust dnn embeddings for speaker recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5329–5333.

Wang, H., Wang, Y., Zhou, Z., Ji, X., Gong, D., Zhou, J., Li, Z., and Liu, W. (2018). Cosface: Large margin cosine loss for deep face recognition.