HOW KNN IMPROVES THE RELIABILITY OF NEURAL MODELS?

Xie ZiCheng, Cui Peng, Jiang WeiBo, Xi HuaJun

ABSTRACT

The reliable measurement of confidence in classifiers' predictions is very important for many applications, and is therefore an important part of classifier design. A promising approach is to design a *confidence score* which measures the confidence of the model when making predictions. However, prior methods impose a strong distributional assumption of the underlying feature space, which may not always hold. In this paper, we explore the efficacy of non-parametric nearest-neighbor distance for computing confidence score. We empirically show that the KNN-based confidence score is a suitable indicator of prediction accuracy: on a relatively large-scale dataset, a higher KNN score means models are more likely to be incorrect. To further verify the efficacy of our confidence score, we evaluate the KNN score under the setting of out-of-distribution detection, and results highlight that the non-parametric KNN method shows promise in addressing OOD detection challenges.

1 INTRODUCTION

Machine learning is being deployed in many high-stakes tasks, such as autonomous driving (Badue et al., 2021), medical diagnostics (Caruana et al., 2015), and financial decision-making (Kaastra & Boyd, 1996). The trust and safety in these applications are critical, as any erroneous prediction can be costly and dangerous. Therefore, a trustworthy model should not only be accurate but be aware of when they are likely to be incorrect. A promising approach can be designing a *confidence score* which measures the confidence of the model when making predictions. More specifically, a proper confidence score should quantify the relative position of a given data sample with respect to decision boundaries, thereby indicating the accuracy of the model's prediction.

A rich line of confidence score algorithms leverages representation embeddings extracted from a model, and operates under an assumption: if a data sample is far from the training data, the model will struggle to cope with it because it is not well-represented in the training distribution. For example, (Lee et al., 2018) modeled the feature embedding space as a mixture of multivariate Gaussian distributions, and used the maximum Mahalanobis distance as a confidence score. However, all these approaches make a strong distributional assumption of the underlying feature space being class-conditional Gaussian. Motivated by recent works (Yuksekgonul et al., 2023; Sun et al., 2022; Mandelbaum & Weinshall, 2017; Jiang et al., 2018), we address this limitation by employing the non-parametric nearest neighbor to compute a confidence score.

In this project, we explore how to employ K Nearest Neighbors(KNN) to solve a confidence score. Specifically, we find the k-th nearest neighbor (KNN) between the embedding of a test input and the embeddings of the training set, and compute the mean of the Euclidean distances between the test embedding and neighbor embeddings. We then treat this KNN distance as a confidence score that implies the accuracy of sample predictions.

Our experiments show that the KNN-based confidence score is a suitable indicator of prediction accuracy: on a relatively large-scale dataset, a higher KNN score means models are more likely to be incorrect. To further verify the efficacy of our confidence score, we evaluate the KNN score under the setting of out-of-distribution detection where confidence is used to predict whether a test point belongs to in-distribution(ID) sample or out-of-distribution(OOD) sample. Results show that the non-parametric KNN method shows promise in addressing OOD detection challenges.

2 RELATED WORKS

Uncertainty qualification. The Bayesian approach seeks to compute a posterior distribution over the parameters of the neural network which is used to estimate prediction uncertainty, as in (Mackay, 1992) and (Neal, 2012). However, Bayesian neural networks are not always practical to implement, and the computational cost involved it typically high. In accordance, in a method which is referred to below as MC-Dropout, (Gal & Ghahramani, 2016) proposed to use dropout during test time as a Bayesian approximation of the neural network, providing a cheap proxy to Bayesian Neural Networks. (Lakshminarayanan et al., 2017) proposed to use Adversarial Training to improve the uncertainty measure of the entropy score of the neural network.

Confidence score. Still, the most basic and one of the most common confidence scores for neural networks can be derived from the strength of the most activated output unit, or rather its normalized version (also called softmax output or max margin). A confidence score that handles better a situation where there is no one class which is most probable, is the (negative) entropy of the normalized network's output. (Zaragoza & d'Alché Buc, 1998) compared these scores, as well as some more complex ones (Tibshirani, 1996), demonstrating somewhat surprisingly the empirical superiority of the two most basic methods described in the previous paragraph.

OOD detection. Modern machine learning models often struggle with out-of-distribution (OOD) inputs samples from a different distribution that the network has not been exposed to during training, and therefore should not be predicted at test time. This gives rise to the importance of OOD detection, which determines whether an input is ID or OOD and enables the model to take precautions. One line of work attempted to perform OOD detection by devising scoring functions (Bendale & Boult, 2015; Chen et al., 2020; Liang et al., 2017). Another promising line of work addressed OOD detection by training-time regularization (Lee et al., 2017; Mohseni et al., 2020; Katz-Samuels et al., 2022).

KNN for anomaly detection. KNN has been explored for anomaly detection (Tian et al., 2014; Zhao & Lai, 2022; Bergman et al., 2020), which aims to detect abnormal input samples from one class. Some other recent works (Dang et al., 2015; Gu et al., 2019; Pires et al., 2020) explore the effectiveness of KNN-based anomaly detection for the tabular data. Motivated by these works, our project explores the potential of KNN to computing confidence score.

3 KNN-BASED CONFIDENCE SCORE

Notation. We consider a standard supervised classification problem where the input space and the label space with K classes are denoted as $\mathcal{X}$ and $\mathcal{Y} := \{1, 2, \cdots, K\}$, respectively. We suppose a data set with N samples is given as $\mathcal{I} := \{\boldsymbol{x}_i, y_i\}_{i=1}^N$, sampled $i.i.d$ from a joint data distribution $\mathcal{P}_{\mathcal{X}\mathcal{Y}}$. Let $f : \mathcal{X} \to \mathbb{R}^K$ denote a classifier, which maps an input to an output space. Let $(\boldsymbol{x}, y) \sim \mathcal{P}_{\mathcal{X}\mathcal{Y}}$ be a random data pair and $\boldsymbol{f}_y(\boldsymbol{x})$ denotes the y-th element of logits vector $\boldsymbol{f}(\boldsymbol{x})$ corresponding to the ground-truth label y. The conditional probability of class y can be approximated by softmax probability $\pi_y(\boldsymbol{x})$, where

$$\pi_y(\boldsymbol{x}) = \sigma(f(\boldsymbol{x}))_y = \frac{e^{f_y(\boldsymbol{x})}}{\sum_{i=1}^K e^{f_i(\boldsymbol{x})}}.$$

where σ is a softmax function. Let the softmax probability distribution be

$$\boldsymbol{\pi}(\boldsymbol{x}) = (\pi_1(\boldsymbol{x}), \pi_2(\boldsymbol{x}), \cdots, \pi_K(\boldsymbol{x}))$$

Then, the classification prediction is given by $\hat{y} = \operatorname{argmax}_{y \in \mathcal{Y}} \pi_y(\boldsymbol{x})$.

Confidence score. In real-world decision-making, the models should not only be accurate but also need to indicate when they are likely to be incorrect. A prevailing assumption in the trustworthy machine learning area is that if a data sample is *atypical* with respect to the training data, the model will be less likely to cope with it well because it is under-represented in the training distribution. Here, we give a formal definition to this *atypicality*:

Definition 3.1 *For any data sample $(\boldsymbol{x}, y)$, the atypicality of this input is defined as*

$$a_{\mathcal{X}}(\boldsymbol{x}) = -\max_y \log P_{(\boldsymbol{x},y)\sim\mathcal{P}_{\mathcal{X}\mathcal{Y}}}\{X = \boldsymbol{x} | Y = y\}$$

We use the logarithm of the class-conditional densities due to high dimensionality and density values being close to zero. A *confidence score* is a function $\mathcal{S} : \boldsymbol{x} \to \mathbb{R}$ that maps any input $\boldsymbol{x}$ to a real value, implying whether the data sample is atypical. From this perspective, a high (or low) confidence score should correspond to a high probability of disagreement with the Bayes-optimal classifier.

3.1 METHOD

In this section, we describe our approach using the deep K Nearest Neighbor (KNN) for computing confidence score, which can be categorized as a distance-based method. The distance-based method leverages representation embeddings extracted from a model and operates under an assumption: if a data sample is far from the training data, the model will struggle to cope with it because it is not well-represented in the training distribution.

Specifically, we compute the k-th nearest neighbor distance between the embedding of each test image and the training set. Importantly, we use the normalized penultimate feature $\boldsymbol{z} = \phi(\boldsymbol{x})/||\phi(\boldsymbol{x})||_2$ for solving confidence score, where $\phi : \boldsymbol{x} \to \mathbb{R}^m$ is a feature encoder. Denote the embedding set of training data as $\boldsymbol{Z} = (\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_n)$. During testing, we derive the normalized feature vector $\boldsymbol{z}_{\text{test}}$ for a test sample $\boldsymbol{x}_{\text{test}}$, and calculate the Euclidean distances $d_i = ||\boldsymbol{z}_i - \boldsymbol{z}_{\text{test}}||_2$ with respect to embedding vectors $\boldsymbol{z}_i \in \boldsymbol{Z}$. Denote the reordered Euclidean distances $(d_1, d_2, \cdots, d_n)$ as $\boldsymbol{D}_{\text{test}} = (d_{(1)}, d_{(2)}, \cdots, d_{(n)})$. The corresponding confidence score is given by:

$$\mathcal{S}(\boldsymbol{x}_{\text{test}}) = \frac{1}{k} \sum_{i=1}^{k} d_{(i)}$$

We proceed to show the efficacy of the KNN-based confidence score in the next section.

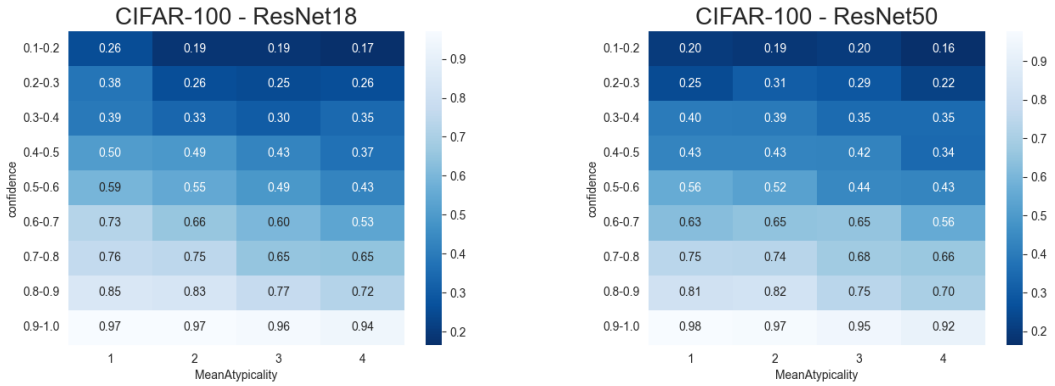
4 EXPERIMENTS

The goal of our experimental evaluation is to answer the following questions: **can the KNN-based confidence score indicate the prediction accuracy?** We use two widely-used datasets: CIFAR10 and CIFAR100. We employ two models: ResNet18 and ResNet50, and we train these models from scratch. We use the standard split with 50,000 training images and 10,000 test images. To illustrate the efficacy of KNN-based confidence score, we compute the KNN score for each test sample, and then we group them according to their scores and confidences (i.e. $\max_{y \in \mathcal{Y}} \pi_y(\boldsymbol{x})$). We will show that in some circumstances, within the same confidence range, predictions for points which have high KNN scores have lower accuracies.

Additionally, we evaluate the confidence score under the setting of out-of-distribution(OOD) detection where confidence is used to predict whether a test point belongs to in-distribution(ID) sample or OOD sample. A line of works attempted to detect OOD sample with a devised confidence score since an OOD sample can be regarded as an atypical sample. Therefore, an appropriate confidence score should assign a high (or low) value to ID samples compared with OOD samples. We will verify that the KNN-based confidence score is capable of distinguishing ID data against OOD data. We use CIFAR10 as ID dataset and SVHN as OOD dataset. We will train ResNet18 and ResNet50 on CIFAR10 and compare the KNN score of ID data and OOD data.

4.1 IS KNN-BASED CONFIDENCE SCORE A PROPER CONFIDENCE SCORE?

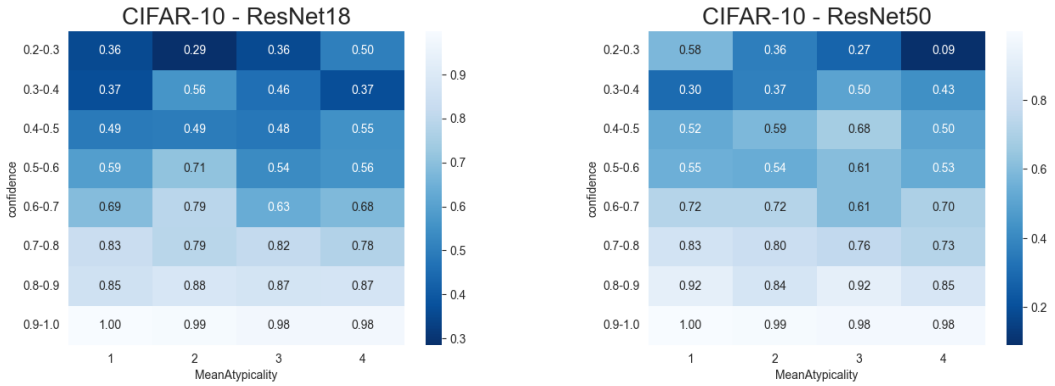
KNN-based confidence score is a proper indicator of prediction accuracy on the large-scale dataset. On CIFAR100, we compare the accuracy of data samples with different KNN scores and confidences. The results are reported in Figure 1, exhibiting that KNN-based confidence score works well on CIFAR100: for the data samples in the same confidences range, as KNN scores increase, the accuracy of sample predictions generally drops. For example, on ResNet50, for data sample within confidence 0.1-0.2, with the increase of KNN scores, the origin accuracy 0.26 in group 1 declines to 0.19 in group 2, 0.19 in group 3 and 0.17 in group 4; for data sample whose confidence is in 0.6-0.7, the accuracy reduces from 0.73 to 0.66, 0.60, and 0.53. This observation highlights that KNN score can serve as a suitable confidence score: a high KNN score indicates more uncertainty.



(a) CIFAR10 - ResNet18 (b) CIFAR10 - ResNet50

Figure 1: Comparison study of the accuracy for data sample with different KNN scores and confidence.

KNN-based confidence score fails to indicate accuracy on the small-scale dataset. We group the test data sample according to their scores and confidences, and we compute the accuracy of predictions for each intersection. The results on CIFAR10 are demonstrated in Figure 2, showing that the relationship that as the KNN score increases, the accuracy of sample predictions decreases does not exist. For example, on ResNet18, we notice that for those samples whose confidences are among 0.2-0.3, as KNN scores increase, the accuracy of the sample predictions first drops from 0.36 to 0.29 in group 1 and group 2, but rebounds to 0.36 and 0.50 in group 3 and group 4. However, this seemingly discouraging outcome may be expected: CIFAR10 is a relatively small-scale dataset, and thus contains few atypical data samples. Therefore, the KNN-based confidence score fails to indicate accuracy under this circumstance since data samples are similar concerning KNN scrores.



(a) CIFAR100 - ResNet18 (b) CIFAR100 - ResNet50

Figure 2: Comparison study of the accuracy for data sample with different KNN scores and confidence.

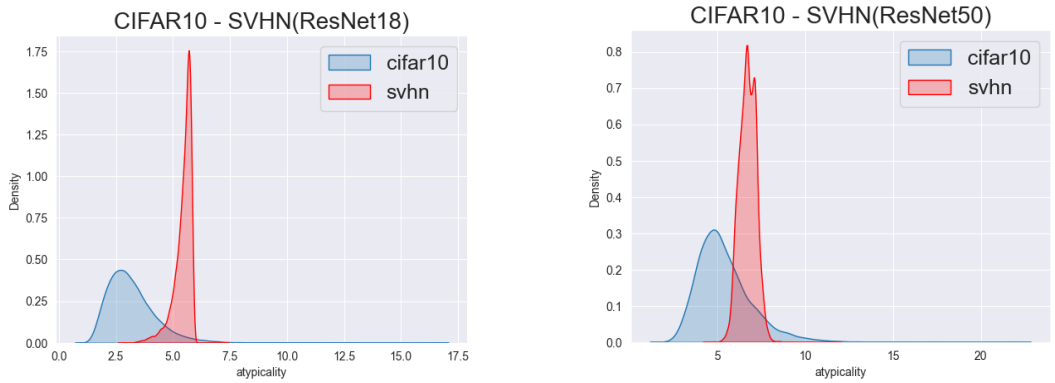
In conclusion, for a diversified dataset (e.g. CIFAR100) where models are more susceptible to atypical samples, a higher KNN-based confidence score means models are more likely to be incorrect; for a simple dataset (e.g. CIFAR10) where the atypical data sample insignificantly influences models' performance, KNN-based confidence may fail to indicate accuracy.

4.2 Can KNN-based confidence score be applied to OOD detection?

KNN scores of ID samples and OOD samples exhibit different distributions. We compare the KNN scores of ID data and OOD data in Figure 3. The results show that for OOD samples, the KNN

scores' probability density concentrates around a higher number compared with OOD samples. For example, on ResNet18, the confidence scores for ID data range around 0 to 5, but for OOD data, the confidence scores typically range from 5 to 6. This finding indicates that the KNN-based confidence score is capable of OOD detection (for instance, use a threshold-based criterion to determine if the input is OOD or not).

The model with better generalization capability struggles to detect OOD samples. In Figure 3, we compare the KNN-score computed on ResNet18 and ResNet50. The results exhibit that the ResNet50 has larger overlap area between ID and OOD's confidence score than the ResNet18. This difference may stem from the fact that ResNet50 enjoys better generalization capability than ResNet18, and thus has difficulty separating ID samples against OOD samples.



(a) ResNet18

(b) ResNet50

Figure 3: Comparison of KNN scores for ID samples and OOD samples.

In a nutshell, our findings suggest that the non-parametric KNN method shows promise in addressing OOD detection challenges. However, further comprehensive research is necessary to establish its effectiveness and generalizability across diverse datasets and models.

5 Conclusion

In this project, we explore how to employ K Nearest Neighbors(KNN) to solve a confidence score. Specifically, we find the k-th nearest neighbor (KNN) between the embedding of a test input and the embeddings of the training set, and compute the mean of the Euclidean distances between the test embedding and neighbor embeddings as a KNN-based confidence score. Our experiments show that on a relatively large-scale dataset, a higher KNN score means models are more likely to be incorrect. Under the setting of OOD detection, our results highlight that our confidence score shows promise in addressing OOD detection challenges.

References

Claudine Badue, Rânik Guidolini, Raphael Vivacqua Carneiro, Pedro Azevedo, Vinicius B Cardoso, Avelino Forechi, Luan Jesus, Rodrigo Berriel, Thiago M Paixao, Filipe Mutz, et al. Self-driving cars: A survey. *Expert Systems with Applications*, 165:113816, 2021.

Abhijit Bendale and Terrance Boult. Towards open world recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1893–1902, 2015.

Liron Bergman, Niv Cohen, and Yedid Hoshen. Deep nearest neighbor anomaly detection. *arXiv preprint arXiv:2002.10445*, 2020.

Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In

Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining, pp. 1721–1730, 2015.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Taurus T Dang, Henry YT Ngan, and Wei Liu. Distance-based k-nearest neighbors outlier detection method in large-scale traffic data. In *2015 IEEE International Conference on Digital Signal Processing (DSP)*, pp. 507–510. IEEE, 2015.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059. PMLR, 2016.

Xiaoyi Gu, Leman Akoglu, and Alessandro Rinaldo. Statistical analysis of nearest neighbor methods for anomaly detection. *Advances in Neural Information Processing Systems*, 32, 2019.

Heinrich Jiang, Been Kim, Melody Guan, and Maya Gupta. To trust or not to trust a classifier. *Advances in neural information processing systems*, 31, 2018.

Iebeling Kaastra and Milton Boyd. Designing a neural network for forecasting financial and economic time series. *Neurocomputing*, 10(3):215–236, 1996.

Julian Katz-Samuels, Julia B Nakhleh, Robert Nowak, and Yixuan Li. Training ood detectors in their natural habitats. In *International Conference on Machine Learning*, pp. 10848–10865. PMLR, 2022.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. *arXiv preprint arXiv:1711.09325*, 2017.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018.

Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690*, 2017.

David John Cameron Mackay. *Bayesian methods for adaptive models*. California Institute of Technology, 1992.

Amit Mandelbaum and Daphna Weinshall. Distance-based confidence score for neural network classifiers. *arXiv preprint arXiv:1709.09844*, 2017.

Sina Mohseni, Mandar Pitale, JBS Yadawa, and Zhangyang Wang. Self-supervised learning for generalizable out-of-distribution detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 5216–5223, 2020.

Radford M Neal. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 2012.

Catarina Pires, Marília Barandas, Letícia Fernandes, Duarte Folgado, and Hugo Gamboa. Towards knowledge uncertainty estimation for open set recognition. *Machine Learning and Knowledge Extraction*, 2(4):505–532, 2020.

Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. In *International Conference on Machine Learning*, pp. 20827–20840. PMLR, 2022.

Jing Tian, Michael H Azarian, and Michael Pecht. Anomaly detection using self-organizing maps-based k-nearest neighbor algorithm. In *PHM society European conference*, volume 2, 2014.

Robert Tibshirani. A comparison of some error estimates for neural network models. *Neural computation*, 8(1):152–163, 1996.

Mert Yuksekgonul, Linjun Zhang, James Zou, and Carlos Guestrin. Beyond confidence: Reliable models should also consider atypicality. *arXiv preprint arXiv:2305.18262*, 2023.

Hugo Zaragoza and Florence d'Alché Buc. Confidence measures for neural network classifiers. In *Proceedings of the Seventh Int. Conf. Information Processing and Management of Uncertainty in Knowlegde Based Systems*, volume 9. Citeseer, 1998.

Puning Zhao and Lifeng Lai. Analysis of knn density estimation. *IEEE Transactions on Information Theory*, 68(12):7971–7995, 2022.