Uncertainty Fingerprints: Interpreting Model Decisions with Human Conceptual Hierarchies

Angie Boggust [1] **Hendrik Strobelt** [2] **Arvind Satyanarayan** [1]

Abstract

Understanding machine learning model uncertainty is essential to comprehend model behavior, ensure safe deployment, and intervene appropriately. However model confidences treat the output classes independently, ignoring relationships between classes that can reveal reasons for uncertainty, such as model confusion between related classes or an input with multiple valid labels. By leveraging human knowledge about related classes, we expand model confidence values into a hierarchy of concepts, creating an *uncertainty fingerprint*. An uncertainty fingerprint describes the model's confidence in every possible decision, distinguishing how the model proceeded from a broad idea to its precise prediction. Using hierarchical entropy, we compare fingerprints based on the model's decision-making process to categorize types of model uncertainty, identify common failure modes, and update dataset hierarchies.

1. Introduction

As machine learning (ML) models are deployed in high-stakes applications (Esteva et al., 2017), it is increasingly important to understand when and why they are uncertain. Uncertainty estimation techniques, such as Bayesian Neural Networks (Abdar et al., 2021), ensembling (Lakshminarayanan et al., 2017), dropout (Gal & Ghahramani, 2016), and calibration (Zadrozny & Elkan, 2001), approximate a model's certainty in its decision and represent this information as a probability distribution. These confidence values provide context about the model's decision-making process (e.g., uncovering when a correct prediction is just a random guess) and are widely used to analyze model behavior and increase human trust in ML systems (Bussone et al., 2015).

[1]MIT CSAIL [2]IBM Research. Correspondence to: Angie Boggust <aboggust@csail.mit.edu>.

AI & HCI Workshop at the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. 2023.

While confidence probability distributions expand human insight into model decision-making, they ignore the relationships between classes that can reveal reasons for uncertainty and inform human intervention. For example, in an autonomous driving setting, a developer might be willing to accept a model that confuses types of vehicles, like `bicycle` and `motorcycle`, since they are treated the same under traffic law. But, they may be wary of a model that regularly confuses `bicycle` with `oak tree`. While these models' probability vectors would have the same distribution and both mistakes would incur the same loss penalty, one is more acceptable because it aligns with human intuition. Currently, interpreting these differences requires manually comparing the model's confidence to human ground truth knowledge. Existing techniques have attempted to explain these differences, but often require additional interpretability methods (Boggust et al., 2022; Kumar et al., 2019) or focus only on data-centric errors (Northcutt et al., 2021). Further, manual analysis can be time-consuming and prohibitive in settings where interpreters do not have ground truth knowledge (e.g., specialized scientific tasks) or ground truth knowledge does not exist (e.g., unknown biological principles).

To overcome these limitations, we expand probability distributions into hierarchical representations of model confidence called *uncertainty fingerprints* (Figure 1). Uncertainty fingerprints leverage the conceptual hierarchies built into many ML datasets (Krizhevsky & Hinton, 2009) that describe parent/child relationships between the output classes and more abstract concepts. To create an uncertainty fingerprint, we propagate uncertainty values through the conceptual hierarchy, expanding the number and complexity of concepts we can describe the model's confidence in. We demonstrate how uncertainty fingerprints can expose model differences obscured by probability distributions, revealing reasons for the model's uncertainty and informing appropriate human intervention (Section 3.1). To analyze uncertainty fingerprints at scale and generate a global understanding of model uncertainty, we introduce hierarchical entropy. Hierarchical entropy encodes the uncertainty distribution at each level of the hierarchy, representing how confident the model is at different levels of complexity. Combining uncertainty fingerprints with hierarchical entropy allows users

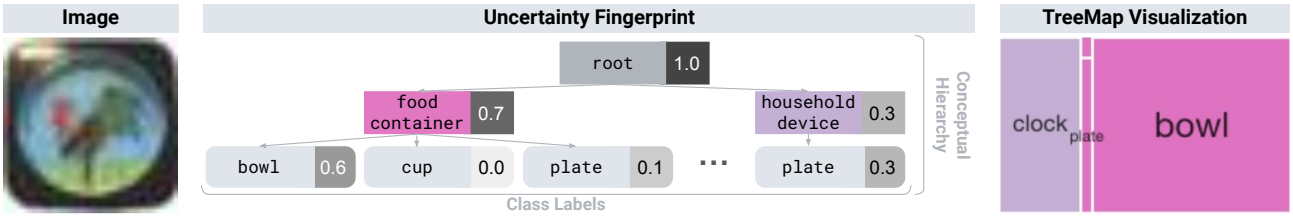

Figure 1. Uncertainty fingerprints reveal the model's decision-making process. Given an input (left), we propagate the model's confidence through a human conceptual hierarchy to create an uncertainty fingerprint (middle) and visualize it as a treemap (right).

to efficiently perform critical model analysis tasks, including categorizing types of model uncertainty and identifying recurring failure modes (Section 3.2).

2. Uncertainty Fingerprints

To generate uncertainty fingerprints, we take in a dataset with a conceptual hierarchy (e.g., CIFAR-100 (Krizhevsky & Hinton, 2009)) and a classification model that outputs a probability distribution over the classes — i.e., a d-dimensional vector, C, representing that model's confidence (c_i) that the input is a member of class i.

2.1. Linking Conceptual Hierarchies to Model Outputs

To encode the relationships between class labels, we combine model probabilities with human conceptual hierarchies. We represent conceptual hierarchies as directed acyclic graphs (DAGs). In each DAG, the leaves are the output classes, the root is an abstract concept, and every path from the root to a leaf is the same length. The DAG consists of m nodes: $N = \{n_0, ..., n_{m-1}\}$. The nodes are split into h levels: $L = \{l_0, ..., l_{h-1}\}$, where the leaves are contained in l_0 (such that $|l_0| = d$). For CIFAR-100 (Krizhevsky & Hinton, 2009), the DAG corresponds to its built-in hierarchy ($h = 3$, $m = 121$, $|l_0| = 100$), where the leaves are the classes, the second level contains the superclasses, and the root is an abstract node connecting all superclasses.

To create *uncertainty fingerprints*, we propagate the model's confidence through the conceptual hierarchy. We compute the model's output probability vector $C = \{c_0, ..., c_d\}$ by applying a softmax to the model's output, such that $\sum C = 1$. Using these confidence values, we assign a fingerprint confidence score f_i to every node n_i in the DAG. The fingerprint confidence of a node is the sum of the model's confidence of every reachable leaf node.

$$f_i = \sum \{c_j \,\forall\, n_j \in l_0 \mid \text{a path exists from } n_i \text{ to } n_j\}$$

Since leaf nodes can only reach themselves, they inherit the model's confidence in their class ($f_i = c_i \,\forall\, n_i \in l_0$).

The resulting uncertainty fingerprint is a set of confidence scores for every concept in the hierarchy. Since the confidences are aggregated via summation, each node's confidence ignores confusion between its children, allowing users to better communicate about the source of a model's uncertainty, such as a level of abstraction confusion (e.g., "*The model is uncertain whether the image is a* `hamster` *or a* `mouse`*, but it is confident it is a* `rodent`". And, they can flag when the model is entirely uncertain and should trade off decision making power with a human (e.g., "*The model is confused until the* `root` *node.").*

2.2. Encoding Fingerprints with Hierarchical Entropy

A common interpretability task is understanding decision-making patterns that recur across an entire dataset, such as identifying common mistakes (Wexler et al., 2019) or categorizing decisions (Boggust et al., 2022). To analyze uncertainty fingerprints across entire datasets and generate a global understanding of model behavior, we introduce hierarchical entropy metrics that measure the similarity of uncertainty fingerprints.

One method to measure fingerprint similarity is to perform a pair-wise comparison of the uncertainty at each node; however, this metric is equivalent to comparing the model's output probabilities where two fingerprints are similar if they have similar confidence in the same classes. Instead, to leverage the added information in the conceptual hierarchy, we measure similarity using hierarchical entropy. Hierarchical entropy encodes how the model's uncertainty changes as it proceeds from a broad idea (root) to a precise concept (leaf), ignoring the model's semantic categorization of an input. With hierarchical entropy, two semantically different inputs can be highly similar if the model is similarly confident about their sibling and parent nodes.

Hierarchical entropy measures the entropy of the model's confidence for each level in the conceptual hierarchy. It takes in an uncertainty fingerprint and outputs a vector of length h representing the uncertainty at each level.

$$\text{HH}_k(F) = -\sum \{f_i * log(f_i) \,\forall\, f_i \in F \mid n_i \in l_k\}$$

Given an uncertainty fingerprint can be thought of as levels of probability distributions, hierarchical entropy measures how certain the model is at each level of complexity. If the

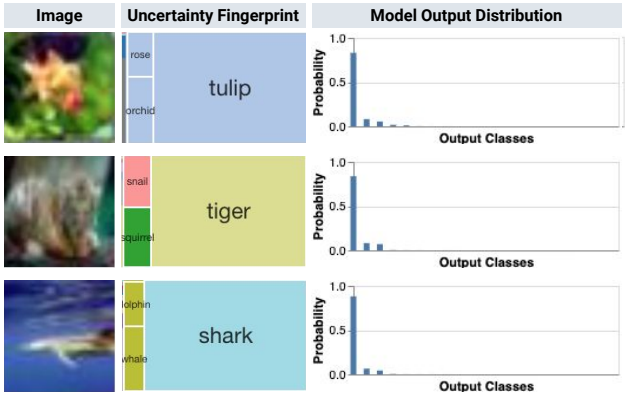

Figure 2. Uncertainty fingerprints reveal reasons for model uncertainty that confidence distributions obscure, such as low-level confusion (top), high-level confusion (mid), and dataset issues (bottom).

model is fully confident in a single node, then that level's entropy (and the entropy of the levels above it) will be 0. But, if the model is very uncertain and its confidence is distributed across many nodes in a level, that level's entropy (and the entropy of the levels below it) will be high.

3. Interpretability Workflows

Using uncertainty fingerprints in an interpretability workflow can help users characterize model decisions, identify patterns of model confusion, and rethink existing human conceptual hierarchies. We demonstrate how to use uncertainty fingerprints to interpret a ResNet50 model (He et al., 2016) trained on CIFAR-100 (Krizhevsky & Hinton, 2009).

3.1. Characterizing Model Decisions

By expanding model uncertainty into a conceptual hierarchy, uncertainty fingerprints can characterize model decisions more precisely than output confidence alone. For example, the images in Figure 2 have nearly identical probability distributions. The model classifies each image with approximately 80% confidence and splits the remaining confidence between two other classes. Using only the probability distribution, these images seem to have identical uncertainty. However, with uncertainty fingerprints, we have a more detailed description of the model's confusion, allowing us to identify that the model is uncertain about these three images for distinctly different reasons.

In the top image, the model is certain the image contains a `flower`, but unsure which species (`tulip`, `orchid`, or `rose`). Originally, a model developer may have been disappointed in our model's uncertain predictions, but knowing it is confident in the higher-level concept `flower`, they may be more tolerant of its low-level confusion. Despite having a similar probability distribution, the uncertainty fingerprint of the middle image reveals a different type of confusion. The model is confused between `tiger`, `squirrel`, and `snail`, and it does not become highly confident until the `root` node. The developer may be more worried about this type of confusion since a human could easily distinguish this image as a `tiger`, so they may explore if it is a one-off confusion or if the model is regularly confused about `tigers` and they should consider adding more data. Finally, the bottom fingerprint reveals a case of uncertainty that could inform new conceptual hierarchies. The model is confused between `shark`, `dolphin`, and `whale`. While all three animals are visually similar, the CIFAR-100 conceptual hierarchy is based on biological taxonomies, so `shark` is part of the `fish` subtree while `dolphin` and `whale` are `aquatic mammals`. This hierarchical structure may not be suitable for an image classification task, where the model relies on visual similarity, so the developer may consider restructuring the hierarchy itself.

Without uncertainty fingerprints, distinguishing these cases of uncertainty would have required manually inspecting the class outputs and mentally measuring their similarity. However, these differences immediately appear with uncertainty fingerprints, allowing users to quickly interpret their model's uncertainty and make an appropriate intervention, like adding more training data, updating the human hierarchy, or accepting some types of confusion.

3.2. Identifying Patterns of Model Confusion

Using uncertainty fingerprints to characterize model decisions in Section 3.1 begs the question, *what types of uncertainty does a model experience the most?* To reveal global patterns of model confusion, we generate uncertainty fingerprints for every image in the CIFAR-100 test dataset and measure their similarity using hierarchical entropy. Since the hierarchical entropy vector is interpretable (i.e., each item represents the entropy at that level), we can generate hierarchical entropy vectors that represent a type of uncertainty and use them to query the uncertainty fingerprints. Given a query vector, we return all the uncertainty fingerprints within an entropy tolerance (0.1) of the query vector or a range of query vectors. These fingerprints represent images where the model experiences the queried uncertainty type (e.g., low-level or high-level uncertainty).

In Figure 3, we query four types of model confusion: confident (i.e., the model is confident in a single prediction), low-level confusion (i.e., the model is confused between classes in the same subtree), multipath confusion (i.e., the model is split between two classes in different subtrees), and high-level confusion (i.e., the model is confused until the root node). Immediately this analysis reveals that confident cases only make up 29% of the model's decisions, and the

Figure 3. Querying uncertainty fingerprints with hierarchical entropy reveals patterns of model confusion and their frequency, such as confident predictions (26%), low-level confusion (45%), multipath confusion (2%), and high-level confusion (16%).

confident images are often clear, well-framed photos. The model experiences confusion 71% of the time, and the most common confusion is low-level confusion which makes up 49% of model decisions. Knowing the model frequently confuses related classes, like `lobster` and `crab`, can inform human intervention. In settings like image tagging, a model developer may tolerate a model with low-level confusion because a high-level tag (e.g., `invertebrate`) is acceptable for image searching. But, in a high-stakes setting like medical image diagnoses, the developer may want to continue iterating on the model before deployment.

Analyzing patterns of model uncertainty per class reveals that the model makes decisions differently for different classes. For instance, only 2% of `boy` predictions are confident, while 73% are low-level confusion. This pattern persists across other `people` classes (e.g., `girl` and `baby`), indicating that the model is likely to confuse `people` but only with other `people`. Similarly, while the model only experiences high-level confusion 16% of the time, the model is frequently high-level confused about `bears` — 44% of `bears` are confused with at least two other classes from separate subtrees. Understanding these patterns can indicate where to spend model development effort, such as adding additional training data from these classes.

With uncertainty fingerprints, we can more easily describe model behavior and categorize its types of uncertainty, revealing reasons to trust or distrust the model, situations to trade-off model decision making, and areas for model improvements.

4. Discussion

Uncertainty fingerprints operationalize model uncertainty values and provide insight into model behavior. By propagating model confidences through a human conceptual hierarchy, uncertainty fingerprints provide a rich vocabulary to describe model uncertainty. In an interpretability workflow, uncertainty fingerprints reveal the reasons for model uncertainty that are hidden by raw probabilities, indicating a model's trustworthiness, suggesting areas for model improvement, and exposing inconsistencies in dataset hierarchies. By combining uncertainty fingerprints with hierarchical entropy metrics and querying for types of model confusion, we identify recurring patterns of model decision

making that can inform model development.

Uncertainty fingerprints reveal rich areas for future work. For instance, applying uncertainty fingerprints to tasks beyond image classification, such as medical tasks (Chen et al., 2021), visual question answering (Antol et al., 2015), image captioning (Lin et al., 2014), or language-based tasks (Miller, 1998), could expose if these uncertainty patterns persist across domains or if specific models are prone to particular types of uncertainty. Currently applying uncertainty fingerprints requires a human-defined hierarchy. New methods could explore hierarchy generation based on model embeddings or repeated class confusion. With both human and machine-generated hierarchies, uncertainty fingerprints could be used to compare across multiple hierarchies to understand which conceptualization aligns best with a given model's decision making process. Finally, with wider applicability, uncertainty fingerprints could integrate into existing HCI and AI analysis pipelines and visualization systems (Tenney et al., 2020; Wexler et al., 2019) for investigation of model confusion by a broad range of users.

Acknowledgements

This work is supported by a grant from the MIT-IBM Watson AI Lab. Research was also sponsored by the United States Air Force Research Laboratory and the United States Air Force Artificial Intelligence Accelerator and was accomplished under Cooperative Agreement Number FA8750-19-2-1000. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the United States Air Force or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

Abdar, M., Pourpanah, F., Hussain, S., Rezazadegan, D., Liu, L., Ghavamzadeh, M., Fieguth, P. W., Cao, X., Khosravi, A., Acharya, U. R., Makarenkov, V., and Nahavandi, S. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, 76:243–297, 2021.

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C. L., and Parikh, D. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pp. 2425–2433, 2015.

Boggust, A., Hoover, B., Satyanarayan, A., and Strobelt, H. Shared interest: Measuring human-ai alignment to identify recurring patterns in model behavior. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pp. 1–17, 2022.

Bussone, A., Stumpf, S., and O'Sullivan, D. The role of explanations on trust and reliance in clinical decision support systems. In *International Conference on Healthcare Informatics (ICHI)*, pp. 160–169. IEEE Computer Society, 2015.

Chen, P.-F., Wang, S.-M., Liao, W.-C., Kuo, L.-C., Chen, K.-C., Lin, Y.-C., Yang, C.-Y., Chiu, C.-H., Chang, S.-C., Lai, F., et al. Automatic icd-10 coding and training system: deep neural network based on supervised learning. *JMIR Medical Informatics*, 9(8):e23230, 2021.

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., and Thrun, S. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118, 2017.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Balcan, M. and Weinberger, K. Q. (eds.), *International Conference on Machine Learning (ICML)*, volume 48, pp. 1050–1059. JMLR, 2016.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. 2009.

Kumar, R. S. S., Brien, D. O., Albert, K., Viljöen, S., and Snover, J. Failure modes in machine learning systems. *arXiv preprint arXiv:1911.11034*, 2019.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 6402–6413, 2017.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

Miller, G. A. *WordNet: An electronic lexical database*. MIT press, 1998.

Northcutt, C., Jiang, L., and Chuang, I. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70:1373–1411, 2021.

Tenney, I., Wexler, J., Bastings, J., Bolukbasi, T., Coenen, A., Gehrmann, S., Jiang, E., Pushkarna, M., Radebaugh, C., Reif, E., et al. The language interpretability tool: Extensible, interactive visualizations and analysis for nlp models. *arXiv preprint arXiv:2008.05122*, 2020.

Wexler, J., Pushkarna, M., Bolukbasi, T., Wattenberg, M., Viégas, F., and Wilson, J. The what-if tool: Interactive probing of machine learning models. *IEEE transactions on visualization and computer graphics*, 26(1):56–65, 2019.

Zadrozny, B. and Elkan, C. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *International Conference on Machine Learning (ICML)*, pp. 609–616, 2001.