Learning to Defer with Better Features: A Case for Going Beyond the Final Layer

Andrew Ponomarev, Anton Agafonov
St. Petersburg Federal Research Center of the Russian Academy of Sciences (SPC RAS)
St. Petersburg, Russian Federation
ponomarev@iias.spb.su, agafonov.a@spcras.ru

***Abstract*—Learning to defer enables machine learning models to selectively pass uncertain decisions to a human expert, improving reliability in critical settings. Existing approaches typically base deferral policies on the model's final layer, which is optimized for classification rather than calibrated deferral. In this work, we introduce a framework for learning deferral policies using intermediate representations from deep convolutional networks. We evaluate this approach on two benchmark datasets, Galaxy-Zoo and CIFAR-10H, across both learning-based and confidence-based deferral strategies. Our results show that hidden-layer features, particularly from deeper residual blocks, enable more effective deferral decisions and improve the accuracy-coverage trade-off. These findings highlight the value of internal features for selective deferral and motivate future architectures that disentangle classification and deferral components.**

I. Introduction

In high-stakes decision-making domains such as healthcare, finance, and autonomous systems, it is increasingly desirable for machine learning (ML) models to collaborate with human experts rather than operate autonomously. To this end, *learning to defer* (L2D) has emerged as a principled framework that enables models to selectively defer predictions to humans when their own confidence is low or uncertainty is high [1].

Most existing L2D methods derive deferral policies from the final layer of a neural network, typically relying on logits or softmax probabilities as signals of uncertainty [2], [3]. However, these final-layer representations are trained primarily for classification accuracy, not for calibrated deferral.

In contrast, intermediate representations within deep networks capture progressively abstract and hierarchical features of the input [4]. These hidden-layer features may encode information that is more general, diverse, or robust to noise, and thus better suited for supporting deferral decisions. Despite this, their use in L2D has been largely unexplored.

In this work, we investigate whether deferral policies trained on internal features extracted from hidden layers can improve deferral quality over those relying solely on final outputs. Our hypothesis is that leveraging richer internal representations allows for more accurate discrimination between instances that can be confidently handled by the model and those that should be passed to a human expert.

We evaluate this hypothesis empirically using deep convolutional networks trained on two benchmark datasets with human annotations: Galaxy-Zoo and CIFAR-10H. Using a ResNet-18 and ResNet-56 backbone, respectively, we extract features from multiple depths within each network, such as average pooling outputs and residual block activations, and train defer models.

Our experiments show that internal features, particularly from deeper residual layers, consistently lead to better accuracy-coverage trade-offs. Notably, models trained with such features outperform those using only final-layer logits across multiple deferral objectives.

The main contributions of this paper are:

- We introduce a framework for learning deferral policies based on hidden-layer features of deep neural networks.
- We systematically evaluate the impact of different representation depths on deferral performance across two real-world datasets with human labels.
- We demonstrate that intermediate features improve accuracy-coverage metrics in deferral settings.

These findings underscore the importance of feature selection for human-AI collaboration and motivate the development of future architectures that explicitly disentangle classification and deferral components.

II. Related Work

A. Task Allocation in Human-AI Collaboration

Research on human-AI collaboration has extensively explored methods for dynamic task allocation between automated models and human experts. A large body of work focuses on *confidence-based* and *heuristic* deferral strategies, which rely on model uncertainty to decide whether to defer a given prediction.

Early approaches use the model's softmax output as a proxy for confidence [5], [6], deferring predictions when the maximum softmax probability falls below a fixed threshold. More sophisticated uncertainty estimation techniques include Monte Carlo Dropout (MC-Dropout) [7], [8], which performs multiple stochastic forward passes, and model ensembles [9], which rely on disagreement across independently trained models.

Madras et al. [1] proposed a practical framework using dual thresholds (t_0, t_1) to assign predictions to the model or to the human expert. More recent work extends this idea by incorporating human behavior modeling. For example, Raghu et al. [10] train a proxy model to estimate expert disagreement using internal embeddings, while Popat et al. [11] apply

Bayesian modeling to improve deferral robustness and sample efficiency. These methods highlight a key trade-off: heuristic deferral is computationally simple and data-efficient, but less effective when expert behavior varies or is imperfect [12].

B. Learning-Based and Cost-Aware Deferral

Beyond heuristics, recent approaches formulate deferral as a learnable, cost-sensitive decision problem. Mozannar and Sontag [2] introduce a differentiable surrogate loss that includes an explicit "defer-to-expert" class, enabling end-to-end optimization. Verma et al. [3] address limitations in softmax calibration by proposing a one-vs-all (OvA) formulation that yields better uncertainty estimates.

Extensions to multi-expert deferral settings have also been studied, where the model learns not only whether to defer, but to which expert [13], [14].

C. Evaluation of Hybrid Systems

Evaluating human-AI collaboration introduces challenges beyond conventional accuracy metrics. In settings where every instance is reviewed by both model and human (e.g., sequential decision-making), system performance is typically measured using overall accuracy [12], [13], [15]–[17]. However, in selective deferral settings, additional metrics are necessary.

The most common metrics include *coverage* (the proportion of instances for which the model itself produces predictions) and *system accuracy* (the overall accuracy of the human–model team, where instances outside the coverage are handled by the human). Trade-off curves between coverage and accuracy are widely used to assess deferral strategies and threshold sensitivity [3], [18]–[21].

Despite extensive exploration of model outputs and confidence measures, relatively few works investigate the impact of internal representations (e.g., hidden-layer activations) on deferral performance. In this work, we build on the above foundations by evaluating how internal features can improve both learned and heuristic deferral strategies, particularly in hybrid decision systems involving real human labels.

III. Problem Statement

We consider the problem of learning a model that can either predict a label or defer the decision to a human expert. Formally, let $\mathcal{D} = \{(x_i, m_i, y_i)\}_{i=1}^{N}$ be a dataset, where $x_i \in \mathcal{X}$ is an input instance (e.g., an image), $m_i \in \mathcal{Y}$ is the label provided by a human (or aggregated human vote), and $y_i \in \mathcal{Y}$ is the ground-truth class label. In practice, m_i may be noisy or uncertain, reflecting the variability in human decision-making.

A typical L2D model consists of two components:

- a classifier $f : \mathcal{X} \to \mathbb{R}^{|\mathcal{Y}|}$ that outputs scores (logits) over classes,
- a deferral policy $\pi : \mathcal{X} \to \{0, 1\}$ that decides whether to predict ($\pi(x) = 0$) or defer to the human expert ($\pi(x) = 1$).

Let $\hat{y}_i = \arg\max f(x_i)$ be the model's prediction. The final system prediction is defined as:

$$\tilde{y}_i = \begin{cases} \hat{y}_i, & \text{if } \pi(x_i) = 0 \\ m_i, & \text{if } \pi(x_i) = 1 \end{cases}$$

The goal is to train f and π such that the overall accuracy of $\tilde{y}_i$ on the dataset $\mathcal{D}$ is maximized, possibly under constraints on the coverage (i.e., the fraction of examples handled by the model) or the cost of querying human labels.

Most existing approaches design the deferral policy $\pi(x)$ based on the final output logits of $f(x)$, using confidence scores or post-hoc estimates of uncertainty. In this work, we hypothesize that features extracted from intermediate layers of deep networks contain richer information for estimating deferral decisions. Our objective is to study whether policies that leverage deep features beyond the output layer can improve deferral accuracy and better balance between automation and human input.

IV. Proposed Approach

We propose a L2D framework that integrates intermediate features from the backbone classifier to inform deferral decisions more effectively. Unlike traditional methods that rely solely on the final output logits, our model leverages internal feature representations to estimate whether to defer to a human expert. This design is illustrated in Fig. 1.

A. Feature Connectors

To access information from the classifier's internal processing, we tap into activations from intermediate layers, specifically, those immediately following ReLU non-linearities. ReLU activations are chosen because they preserve sparsity and eliminate negative values, which can amplify the interpretability and relevance of active feature patterns. These layers often retain high-level spatial semantics that may be suppressed in deeper fully-connected layers.

Each selected intermediate layer is passed through a *Connector* module that aggregates the spatial information using global pooling strategies. We apply global average pooling (GAP) and global max pooling (GMP) to summarize the activation maps, capturing both general trends and prominent local features. The outputs from GAP and GMP are concatenated to form compact, yet expressive feature vectors representing each layer's state.

B. Deferral Model

The connector outputs from multiple intermediate layers are concatenated into a single feature vector. Additionally, we include the output of the final average pooling layer before classification, ensuring that the defer model has access to both low-level and high-level features.

This combined feature vector is passed to a multi-layer perceptron (MLP), which produces a vector of logits. The first $|\mathcal{Y}|$ components correspond to class predictions (mimicking the main classifier), while the final component represents the

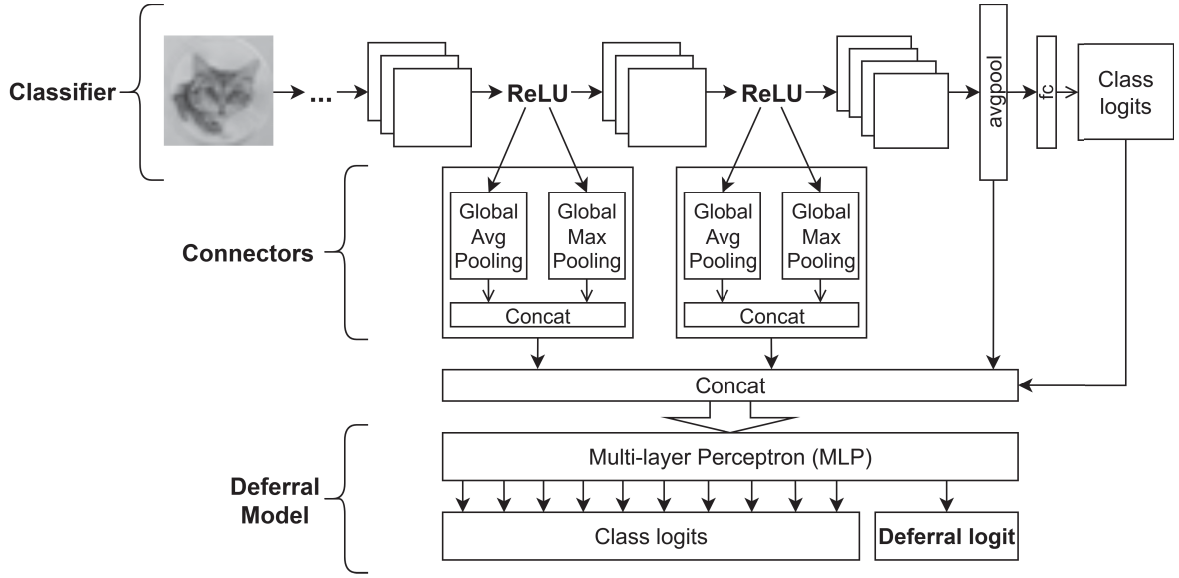

Fig. 1. Overview of the proposed approach. Intermediate features from the classifier are aggregated and passed to a deferral model that decides whether to predict or defer

deferral logit, that is, the model's tendency to defer rather than predict.

The defer decision is then made by thresholding the deferral logit. During training, we optimize both components jointly to maximize the accuracy of the final prediction system, which may defer or classify based on this learned policy.

V. Experiment Setup

A. Datasets

We evaluate our approach on two datasets that provide human annotations and enable studying deferral policies: CIFAR-10H and Galaxy Zoo.

1) CIFAR-10H: CIFAR-10H [22] is an extension of the CIFAR-10 [23] test set, providing soft labels that reflect human perceptual uncertainty in image classification. The dataset includes annotations for the 10,000 images in the CIFAR-10 test set, with 1,000 images for each of the 10 categories.

A total of 511,400 human classifications were collected via Amazon Mechanical Turk, involving 2,571 participants. Workers were asked to categorize each image into one of 10 labels, with the label positions randomized for each trial. They were instructed to classify the images as quickly and accurately as possible, though there was no time limit. After an initial training phase, each participant classified 200 images (20 from each category). To ensure data quality, every 20 images, an easy-to-classify image was presented as an attention check, and participants who scored below 75% on these checks (14 total) were excluded from the final analysis. On average, 51 judgments were collected per image (range: 47–63).

2) Galaxy Zoo: The Galaxy Zoo dataset [24] consists of images of galaxies labeled by citizen scientists through a crowdsourcing platform. Each image is annotated by multiple volunteers who answer a series of morphological questions (e.g., regarding smoothness, presence of features or disks, spiral arms, etc.). The resulting labels form a probability distribution over possible answers, capturing inter-observer uncertainty.

In this work, we used the first 10,000 images from the Galaxy Zoo dataset. We focused on a binary classification task using only the first question: *"Is the object a smooth galaxy, a galaxy with features/disk, or a star?"* Specifically, we selected the following two classes: smooth galaxy; galaxy with features or disk.

B. Classifiers

To perform selective classification and deferral experiments, we first trained baseline classifiers without any built-in mechanism for abstention.

For binary classification on Galaxy-Zoo, we used a ResNet-18 model pretrained on ImageNet. The final fully connected layer was modified to output two logits. Images were processed using standard ImageNet-style transformations, including resizing to 256×256, center cropping to 224×224, and normalization with ImageNet mean and standard deviation. The dataset was limited to the first 10,000 examples (7000 for training and 3000 for testing). The model was trained using the Cross-Entropy loss, Adam optimizer with learning rate 10^{-3}, weight decay 10^{-2}, batch size of 250, and early stopping with a patience of 10 epochs. Training was capped at 100 epochs. The resulting classifier achieved 82.77% accuracy on the test subset.

For the CIFAR-10H dataset, we used a pretrained ResNet-56 model from an open-source repository. The model achieved 94.37% test accuracy without additional fine-tuning. Images were normalized using dataset-specific mean and standard deviation values. This model was used as-is for further deferral experiments.

C. Deferral Approaches

We implemented deferral models using three different approaches:

- *MaxSoftmax:* A simple post-hoc baseline that relies only on the classifier's logits. The maximum softmax probability is used as a confidence score, with low-confidence predictions deferred to the human.
- *L-CE:* A method introduced by Mozannar and Sontag [2], which jointly learns classification and deferral via a combined loss. It treats deferral as an explicit action and optimizes for both accuracy and appropriate rejection, offering better trade-offs than heuristic confidence thresholds.
- *OvA (One-vs-All):* A method introduced by Verma et al. [3], using a one-vs-all surrogate formulation with a single shared deferral logit. This design encourages the model to defer when the expert is more reliable, while producing a unified $K+1$-way output for inference and often improves both coverage and accuracy.

D. Feature Sources for Deferral Models

Unlike standard deferral models that operate solely on final logits, we investigate whether internal network representations can provide richer features for making deferral decisions.

1) Galaxy-Zoo (ResNet-18): We evaluate the following feature sources:

- *Logits:* Output of the final linear classification layer.
- *Avgpool:* Output of the global average pooling layer.
- *Layer4:* All ReLU activations within the final residual block (layer4).
- *Layer3:* All ReLU activations within the penultimate residual block (layer3).
- *Avgpool+Layer4:* Concatenation of avgpool and the ReLU outputs from layer4.

2) CIFAR-10H (ResNet-56): We evaluate four feature sources:

- *Logits:* Output of the final linear classification layer.
- *Avgpool:* The global average pooled representation.
- *Layer3:* The set of ReLU activations from the third residual group.
- *Avgpool+Layer3:* Concatenation of avgpool and the ReLU outputs from layer3.

Here, the term "layer" refers to an entire residual block (e.g., layer3, layer4), and specifically to the set of activations produced by all ReLU operations within that block.

E. Deferral Model Architecture

All deferral models were implemented as multilayer perceptrons (MLPs) with ReLU activations. The output layer always consisted of $k+1$ logits, where k corresponds to the number of classes and the additional logit represents the abstain (defer) option.

The architecture of the MLP is adapted to the dimensionality of each input feature source. Detailed configurations for the Galaxy-Zoo (ResNet-18) and CIFAR-10H (ResNet-56) experiments are provided in Tables I and II, respectively.

TABLE I. MLP CONFIGURATIONS FOR DIFFERENT FEATURE SOURCES ON GALAXY-ZOO (RESNET-18)

Feature Source	MLP Architecture
Logits	[2, 10, 3]
Avgpool	[512, 128, 3]
Layer4	[2048, 512, 3]
Layer3	[1024, 256, 3]
Avgpool + Layer4	[2560, 512, 3]

TABLE II. MLP CONFIGURATIONS FOR DIFFERENT FEATURE SOURCES ON CIFAR-10H (RESNET-56)

Feature Source	MLP Architecture
Logits	[10, 20, 11]
Avgpool	[64, 32, 11]
Layer3	[1152, 256, 11]
Avgpool + Layer3	[1216, 256, 11]

F. Training Procedure

The deferral models were trained using the Adam optimizer with a learning rate of 10^{-3} and a weight decay of 10^{-2}. The batch size was set to 500. Training was performed for a maximum of 500 iterations with early stopping (patience of 20).

1) Galaxy-Zoo: A 5-fold cross-validation setup was used, with 8000 training examples and 2000 test examples in each fold.

2) CIFAR-10H: A hold-out validation strategy was used, with 7000 samples for training and 3000 for testing.

G. Simulating Human Annotations

To simulate noisy human labels for delegated samples, a sampling-based annotation model was used. Given the ground-truth probability distribution over classes, the simulated annotator generated a label by randomly sampling from the distribution. This process captures the inherent stochasticity and variability in human annotations.

H. Evaluation Metrics

To assess the quality of deferral models, we adopt three complementary evaluation metrics: *AC-AUC*, *Maximum Accuracy*, and *Coverage at Maximum Accuracy*. These metrics are computed based on the model's selective predictions—that is, when it chooses to make a prediction itself or defer the decision to a human.

1) Coverage-Accuracy Curve: Each deferral model outputs, for each instance, a vector of logits with $K+1$ values, where K corresponds to the number of target classes, and the last value corresponds to the deferral option. From this, we compute a scalar *deferral score* per instance, indicating how confident the model is in its own prediction relative to deferral. Specifically, we define the score as the difference between the maximum softmax probability among the K classification logits and the softmax probability assigned to the deferral logit.

Using this score, we sort the data points in ascending order, simulating a selective classification process where the model

defers more often for lower scores. At each step, we substitute the model's prediction with the human label and compute the resulting accuracy and remaining model coverage (i.e., fraction of instances the model handles itself). This results in a *coverage-accuracy curve*.

2) AC-AUC: The main evaluation metric is the *Area under the Coverage-Accuracy Curve* (AC-AUC). This metric captures the overall performance of the defer system across the full range of coverage values—from full automation to full deferral. Formally, it is computed as a Riemann sum over the sorted coverage-accuracy points:

$$\text{AC-AUC} = \frac{1}{2} \sum_{i=1}^{N-1} (c_{i+1} - c_i)(a_{i+1} + a_i)$$

where c_i and a_i are the coverage and accuracy values at point i, respectively.

3) Maximum Accuracy and Corresponding Coverage: In addition to AC-AUC, we report the maximum accuracy achieved along the curve and the corresponding coverage value at which it occurs. This helps quantify the point at which optimal overall accuracy is obtained by balancing deferral and automation.

4) Implementation Details: Model predictions are extracted by taking the $\arg\max$ over the first K logits. Human annotations are sampled stochastically from human-provided probability distributions using a fixed random seed. All metrics are computed on held-out validation or test subsets and are averaged across cross-validation folds (for Galaxy-Zoo) or over a single hold-out split (for CIFAR-10H).

VI. Results and Discussion

A. Galaxy-Zoo

We present the results of deferral models trained and evaluated on the Galaxy-Zoo dataset. Fig. 2 displays boxplots summarizing the distribution of three evaluation metrics, AC-AUC, maximum accuracy, and coverage at maximum accuracy, across various feature sources and deferral strategies (L-CE, OvA, MaxSoftmax).

Across all methods, logits prove to be the least informative feature representation, consistently resulting in the lowest AC-AUC scores. In contrast, deeper internal representations such as `avgpool` and `layer4` yield significantly higher AC-AUC values, likely due to their richer feature encoding. `layer3` underperforms, possibly due to its lower-level abstraction. Combining `avgpool` and `layer4` stabilizes performance by reducing the variance of AC-AUC across folds, particularly for L-CE. Among deferral strategies, L-CE consistently outperforms OvA and MaxSoftmax in AC-AUC, indicating better trade-offs between accuracy and coverage. However, OvA often achieves the highest classification accuracy with lower variance. Interestingly, `avgpool` features lead to the highest coverage at peak performance.

These findings highlight the advantages of using deeper internal features, particularly from `avgpool` and `layer4`, for training deferral models. They also suggest that logits, despite their common use, may be suboptimal as the sole input to the deferral head.

B. CIFAR-10H

We further evaluate deferral performance on the CIFAR-10H dataset, with a focus on coverage-accuracy trade-offs. As shown in Fig. 3, the most effective configuration combines `avgpool` and `layer3` features, achieving the highest accuracy and model coverage. Consistent with observations on Galaxy-Zoo, MaxSoftmax again performs the worst across both metrics.

C. General Observations

Across both Galaxy-Zoo and CIFAR-10H, several consistent trends emerge. First, relying solely on final output logits yields the weakest deferral performance, both in terms of accuracy and deferral quality. In contrast, internal feature representations, particularly those extracted after ReLU activations in deeper residual blocks, consistently lead to better results.

Features from the final average pooling layer (`avgpool`) and the deepest residual block (e.g., `layer4` in ResNet-18, `layer3` in ResNet-56) provide the most informative signals for learning effective deferral policies. Combining these sources further improves robustness and stability, particularly by reducing performance variance across folds.

Regarding deferral strategies, L-CE consistently achieves the highest overall trade-off between accuracy and coverage (as measured by AC-AUC), while OvA often obtains the best peak accuracy. MaxSoftmax underperforms in all settings, highlighting its limitations as a confidence-based heuristic.

These findings support the hypothesis that deferral policies benefit from access to rich internal representations. Designing deferral heads that leverage intermediate features can significantly enhance performance over traditional approaches based on output logits alone.

VII. Conclusion

In this work, we studied the impact of using intermediate hidden-layer representations for training deferral policies in human-AI collaboration settings. Although traditional approaches rely on information from the final layer, we demonstrated that deeper internal features, especially those extracted from late stage residual blocks, offer richer and more informative signals for deferral. Across two benchmark datasets, our experiments showed that deferral models trained on such features achieve higher AC-AUC scores, better maximum accuracy, and greater coverage.

Our findings emphasize the underutilized value of deep representations in selective deferral tasks and suggest that future deferral systems should explicitly incorporate such features. Beyond improving performance, this direction encourages the development of architectures that separate decision confidence from final classification, enhancing the interpretability and trustworthiness of human-AI decision pipelines.

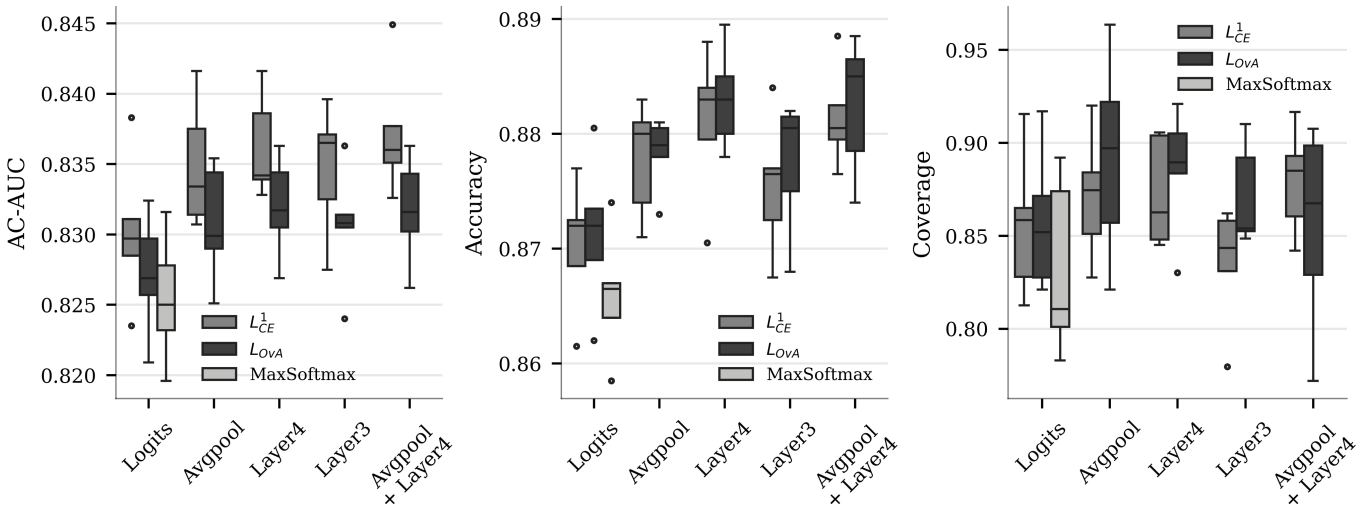

Fig. 2. Comparison of deferral strategies across feature sources on Galaxy-Zoo dataset

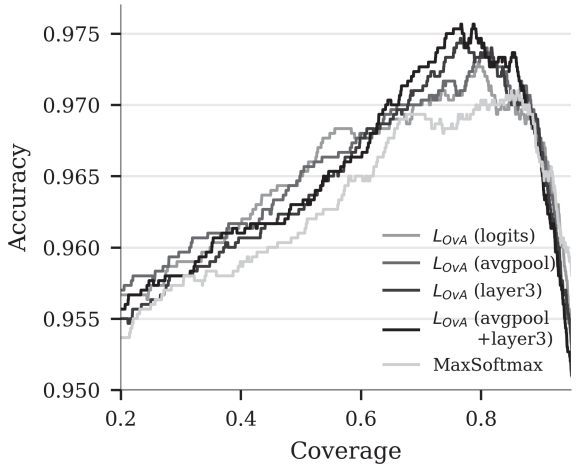

Fig. 3. Accuracy-Coverage curves on CIFAR-10H dataset

ACKNOWLEDGMENT

The research is funded by the Russian Science Foundation (project 24-21-00337).

REFERENCES

[1] D. Madras, T. Pitassi, and R. Zemel, "Predict responsibly: Improving fairness and accuracy by learning to defer," in *Advances in Neural Information Processing Systems*, vol. 2018-December, nov 2018, pp. 6147–6157. [Online]. Available: http://arxiv.org/abs/1711.06664

[2] H. Mozannar and D. Sontag, "Consistent estimators for learning to defer to an expert," *37th International Conference on Machine Learning, ICML 2020*, vol. PartF168147-10, pp. 7033–7044, 2020.

[3] R. Verma and E. Nalisnick, "Calibrated Learning to Defer with One-vs-All Classifiers," in *Proceedings of Machine Learning Research*, vol. 162, feb 2022, pp. 22 184–22 202. [Online]. Available: http://arxiv.org/abs/2202.03673

[4] A. Agafonov and A. Ponomarev, "An Experiment on Localization of Ontology Concepts in Deep Convolutional Neural Networks," in *ACM International Conference Proceeding Series*. New York: ACM, dec 2022, pp. 82–87.

[5] L. P. Cordelia, C. D. Stefano, F. Tortorella, and M. Vento, "A Method for Improving Classification Reliability of Multilayer Perceptrons," *IEEE Transactions on Neural Networks*, vol. 6, no. 5, pp. 1140–1147, 1995.

[6] C. De Stefano, C. Sansone, and M. Vento, "To reject or not to reject: that is the question - an answer in case of neural classifiers," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 30, no. 1, pp. 84–94, 2000.

[7] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," *33rd International Conference on Machine Learning, ICML 2016*, vol. 3, pp. 1651–1660, 2016.

[8] Y. Geifman and R. El-Yaniv, "Selective classification for deep neural networks," *Advances in Neural Information Processing Systems*, vol. 2017-December, pp. 4879–4888, 2017.

[9] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Advances in Neural Information Processing Systems*, vol. 2017-December, pp. 6403–6414, 2017.

[10] M. Raghu, K. Blumer, G. Corrado, J. Kleinberg, Z. Obermeyer, and S. Mullainathan, "The Algorithmic Automation Problem: Prediction, Triage, and Human Effort," mar 2019. [Online]. Available: http://arxiv.org/abs/1903.12220

[11] R. Popat and J. Ive, "Embracing the uncertainty in human–machine collaboration to support clinical decision-making for mental health conditions," *Frontiers in Digital Health*, vol. 5, 2023.

[12] S. Ma, Y. Lei, X. Wang, C. Zheng, C. Shi, M. Yin, and X. Ma, "Who Should I Trust: AI or Myself? Leveraging Human and AI Correctness Likelihood to Promote Appropriate Trust in AI-Assisted Decision-Making," in *Conference on Human Factors in Computing Systems - Proceedings*. New York, NY, USA: ACM, apr 2023, pp. 1–19. [Online]. Available: https://dl.acm.org/doi/10.1145/3544548.3581058https://arxiv.org/abs/2301.05809

[13] R. Verma, D. Barrejón, and E. Nalisnick, "Learning to Defer to Multiple Experts: Consistent Surrogate Losses, Confidence Calibration, and Conformal Ensembles," in *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics, PMLR 206*, 2023, pp. 11 415–11 434. [Online]. Available: http://arxiv.org/abs/2210.16955

[14] A. Mao, M. Mohri, and Y. Zhong, "Principled Approaches for Learning to Defer with Multiple Experts," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 14494 LNCS, 2024, pp. 107–135. [Online]. Available: http://arxiv.org/abs/2310.14774

[15] K. Vodrahalli, T. Gerstenberg, and J. Zou, "Uncalibrated Models Can Improve Human-AI Collaboration," in *36th Conference on Neural Information Processing Systems (NeurIPS 2022)*, 2022. [Online]. Available: http://arxiv.org/abs/2202.05983

[16] P. Hemmer, S. Schellhammer, M. Vössing, J. Jakubik, and G. Satzger, "Forming Effective Human-AI Teams: Building Machine Learning

Models that Complement the Capabilities of Multiple Experts," in *IJCAI International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, jun 2022, pp. 2478–2484. [Online]. Available: http://arxiv.org/abs/2206.07948https://www.ijcai.org/proceedings/2022/344

[17] A. Mao, C. Mohri, M. Mohri, and Y. Zhong, "Two-Stage Learning to Defer with Multiple Experts," in *Advances in Neural Information Processing Systems*, vol. 36, 2023. [Online]. Available: https://openreview.net/forum?id=GIlsH0T4b2

[18] A. De, N. Okati, A. Zarezade, and M. G. Rodriguez, "Classification Under Human Assistance," in *35th AAAI Conference on Artificial Intelligence, AAAI 2021*, vol. 7, 2021, pp. 5905–5913. [Online]. Available: http://arxiv.org/abs/2006.11845

[19] D.-X. Liu, X. Mu, and C. Qian, "Human Assisted Learning by Evolutionary Multi-Objective Optimization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 10, jun 2023, pp. 12 453–12 461. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/26467

[20] M. Kobayashi, K. Wakabayashi, and A. Morishima, "Human+AI Crowd Task Assignment Considering Result Quality Requirements," *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, vol. 9, pp. 97–107, oct 2021. [Online]. Available: https://ojs.aaai.org/index.php/HCOMP/article/view/18943

[21] A. Ponomarev, "A Simple Heuristic for Controlling Human Workload in Learning to Defer," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 15327 LNCS, pp. 120–130, 2025.

[22] J. Peterson, R. Battleday, T. Griffiths, and O. Russakovsky, "Human uncertainty makes classification more robust," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2019-October, pp. 9616–9625, 2019.

[23] A. Krizhevsky, "Learning Multiple Layers of Features from Tiny Images," ... *Science Department, University of Toronto, Tech.* ..., pp. 1–60, 2009. [Online]. Available: http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Learning+Multiple+Layers+of+Features+from+Tiny+Images#0

[24] S. P. Bamford, R. C. Nichol, I. K. Baldry, K. Land, C. J. Lintott, K. Schawinski, A. Slosar, A. S. Szalay, D. Thomas, M. Torki, D. Andreescu, E. M. Edmondson, C. J. Miller, P. Murray, M. J. Raddick, and J. Vandenberg, "Galaxy Zoo: The dependence of morphology and colour on environment," *Monthly Notices of the Royal Astronomical Society*, vol. 393, no. 4, pp. 1324–1352, 2009.