Advanced International Journal for Research (AIJFR)
E-ISSN: 3048-7641 ● Website: www.aijfr.com ● Email: editor@aijfr.com

Design and Implementation of an Explainable Decision-Support Pipeline for Glioma Grading and Brain Tumor MRI Classification

Kavya K[1], Mr. S. Manjunatha[2]

[1,2]Department of Computer Science and Engineering

Abstract
This paper documents the system design and implementation of a two-pipeline clinical decision-support tool for neuro-oncology: a tabular classifier that grades glioma severity from TCGA mutation and clinical records, and a convolutional image classifier that sorts brain MRI scans into four diagnostic categories. Rather than treating predictive accuracy as the only deliverable, we walk through the engineering pipeline that turns trained models into an auditable tool — data ingestion and cleaning, class-imbalance correction, model selection, explanation generation, and web-based deployment. On the tabular side, eight classifier families are trained on 23 engineered features drawn from 839 TCGA patient records; Logistic Regression reaches 86.90% held-out accuracy, and Random Forest attains the top AUC-ROC of 0.9229 (5-fold CV AUC 0.9258). A SHAP TreeExplainer module attached to the XGBoost model produces both global and per-patient feature attributions, surfacing IDH1 mutation status as the dominant predictor, in line with WHO 2021 grading criteria. On the imaging side, ResNet50 and MobileNetV2 are fine-tuned through transfer learning for four-class MRI classification (glioma, meningioma, pituitary, no tumor), reaching up to 95.58% validation accuracy and combined at inference time via soft-vote averaging. Both pipelines sit behind a single Flask web application offering an interactive dashboard for tabular prediction, MRI upload, and real-time SHAP visualization. We describe the module-level architecture, the training and inference workflows, and the evaluation protocol used to validate the system, and close with deployment considerations and current limitations.

Index Terms—Clinical decision-support systems, explainable AI, SHAP, glioma grading, TCGA, transfer learning, ResNet50, MobileNetV2, brain tumor MRI classification, system deployment.

1. INTRODUCTION

Machine-learning papers in oncology frequently report a trained model's accuracy without describing how that model would actually reach a clinician. Yet the gap between a notebook experiment and a usable clinical tool is where most of the engineering effort lives: input validation, preprocessing consistency between training and inference, explanation generation on demand, and a front end a non-programmer can operate. This paper treats that gap as the subject of study rather than an afterthought, using a glioma-grading and brain-tumor-MRI system as the concrete case.

Two data modalities are available for glioma assessment. Molecular profiling — a panel of gene-mutation calls plus basic clinical descriptors — lets a model approximate the grading a pathologist would perform,

and The Cancer Genome Atlas (TCGA) supplies a public dataset built for exactly this purpose. Imaging, in the form of an MRI scan, offers a non-invasive alternative that a deep network can classify by tumor type. Neither modality alone is sufficient in practice, so the implementation described here treats them as parallel pipelines that converge in one deployed application rather than as competing approaches.

Beyond model choice, this paper's contribution is architectural: we specify (i) how raw TCGA and MRI data are transformed into model-ready tensors, (ii) how eight tabular classifiers and two convolutional backbones are trained, validated, and selected, (iii) how a SHAP-based explanation layer is attached to the tabular model so that every prediction ships with a rationale, and (iv) how the resulting artifacts are served through a web application designed for interactive, auditable use rather than one-off inference.

2. SYSTEM OVERVIEW AND DESIGN GOALS

The system is organized around three cooperating layers. A data layer ingests and standardizes the TCGA mutation table and the MRI image corpus. A model layer trains, validates, and persists the eight tabular classifiers, the two CNN backbones, and the SHAP explainer built on top of the XGBoost model. A presentation layer, implemented as a Flask application, accepts new patient records or MRI uploads, routes them through the appropriate trained artifacts, and renders both the prediction and its explanation.

Four design goals shaped these layers. First, predictive quality: each pipeline is benchmarked against a battery of classical or transfer-learned baselines rather than a single model, so that the deployed choice is defensible. Second, auditability: every tabular prediction is accompanied by a SHAP attribution, since a probability alone is not enough for a clinical audience to act on. Third, computational efficiency: pairing ResNet50 with the lighter MobileNetV2 lets the imaging pipeline run on either a workstation or a more constrained edge device without retraining. Fourth, reproducibility: SMOTE-based rebalancing is confined strictly to the training fold, and evaluation always uses a held-out partition that never participates in oversampling or hyperparameter search.

The remainder of this paper follows the system, not the narrative of a single experiment: Section III describes how each data source is prepared; Section IV specifies the model architectures and the algorithms used to train and explain them; Section V details the training protocol; Section VI reports the resulting evaluation; Section VII describes how the trained artifacts are wired into the deployed application; Section VIII situates the work relative to prior studies; and Sections IX–X discuss limitations and conclude.

3. DATA ENGINEERING PIPELINE

A. Molecular Data Preparation

The tabular pipeline consumes the TCGA-derived “Glioma Grading Clinical and Mutation” table: 839 patient rows, each carrying 20 binary gene-mutation flags (IDH1, TP53, ATRX, PTEN, EGFR, CIC, MUC16, PIK3CA, NF1, PIK3R1, FUBP1, RB1, NOTCH1, BCOR, CSMD3, SMARCA4, GRIN2A, IDH2, FAT4, PDGFRA), three clinical fields (Gender, Age at Diagnosis, Race), and the grading label (0 for LGG, 1 for GBM). Categorical clinical fields are encoded to numeric form prior to model fitting, consistent with the input requirements of every classifier in Section IV-A. After row-level cleaning, the cohort is partitioned 80/20 in a stratified manner, yielding 671 training rows and 168 held-out test rows; stratification keeps the LGG:GBM ratio comparable across both partitions.

Because GBM cases are underrepresented relative to LGG in TCGA, the training partition — and only the training partition — is rebalanced with SMOTE prior to model fitting. The test partition is never touched by the oversampling step, which keeps the reported metrics representative of the true class distribution a deployed model would encounter.

B. Imaging Data Preparation

The imaging pipeline draws on the publicly hosted Kaggle "Brain Tumor MRI" collection, comprising 3,264 T1-weighted, contrast-enhanced slices across four diagnostic categories — glioma, meningioma, pituitary tumor, and no-tumor cases — pre-split into 2,870 training images and 394 test images. Every slice is resized to 224 × 224 pixels and normalized using the mean and standard deviation conventionally paired with ImageNet-pretrained backbones, since both ResNet50 and MobileNetV2 are initialized from ImageNet weights. During training only, images are augmented on the fly with random horizontal flips, rotations of up to ±10°, and mild color jitter, widening the effective training distribution without altering the held-out test set.

4. MODEL ARCHITECTURE AND ALGORITHMS

A. Classical Ensemble for Molecular Grading

Eight classifier families are trained under one shared protocol so that the final choice of deployed model is an empirical decision rather than a default: Random Forest [13] (500 trees, square-root feature subsetting), Gradient Boosting (300 trees, learning rate 0.05), XGBoost [14] (500 trees, max depth 6, learning rate 0.05 — also the backbone passed to the explainability module), LightGBM [15] (500 leaf-wise trees), CatBoost [16] (500 iterations, ordered boosting), $\ell 2$-regularized Logistic Regression with feature standardization, an RBF-kernel SVM with probability calibration, and a two-hidden-layer (64, 32 units) ReLU MLP.

Algorithm 1 summarizes the shared training procedure applied to each of the eight classifiers.

Algorithm 1: Molecular Classifier Training

Input: stratified train split T, held-out test split H
1. Fit SMOTE on T only; obtain balanced T'
2. For each of the 8 classifier families:
   a. Run 5-fold stratified cross-validation on T' for model-selection diagnostics
   b. Fit the classifier on the full T'
   c. Score the fitted classifier once on H (accuracy, precision, recall, F1, AUC-ROC)
3. Attach SHAP TreeExplainer to the fitted XGBoost model for downstream attribution

B. Convolutional Architecture for Imaging

Both imaging backbones follow the same fine-tuning recipe. ResNet50 and MobileNetV2 are each initialized with ImageNet-pretrained weights, and their classification heads are replaced with a compact block — a 256-unit dense layer, ReLU activation, dropout at rate 0.3, and a final 4-way softmax. Each network is fine-tuned for 20 epochs with the Adam optimizer (learning rate 10^{-4}) at a batch size of 32, and the epoch with the best validation accuracy is checkpointed as that network's deployed weights.

At inference time the two networks are not used independently; their softmax outputs are averaged into a single ensemble probability vector, as given in Eq. (1):

$$p_{ensemble}(c) = \tfrac{1}{2}\left[\, p_{ResNet50}(c) + p_{MobileNetV2}(c) \,\right] \quad (1)$$

where c indexes the four output classes. This soft-voting scheme requires no additional training and lets the deployed application fall back to either backbone individually if the other is unavailable, which is useful for the edge-oriented MobileNetV2 deployment target discussed in Section VII.

C. Explainability Module

Global and local explanations for the tabular pipeline are produced by SHAP TreeExplainer [7], an exact, polynomial-time implementation of Shapley-value attribution for tree ensembles. For a feature i and a prediction function f over a feature set F, the Shapley value is defined in Eq. (2):

$$\varphi_i = \Sigma_{S \subseteq F \setminus \{i\}} [|S|!(|F|-|S|-1)!/|F|!] \cdot [f(S \cup \{i\}) - f(S)] \quad (2)$$

summing, over every subset S of features that excludes i, the marginal contribution of adding feature i, weighted by the number of feature orderings consistent with that subset. TreeExplainer computes this quantity exactly for tree-based models without the combinatorial cost the general definition implies, which is what makes it practical to compute a fresh per-patient explanation at request time rather than only offline. The module is attached to the fitted XGBoost model and produces two outputs consumed by the deployment layer: a global ranking of mean |SHAP| values across the test set (Fig. 1), and a per-instance waterfall decomposition for any single prediction (Fig. 2).

5. TRAINING PROTOCOL AND OPTIMIZATION

Model selection for the tabular pipeline is governed by 5-fold stratified cross-validation on the SMOTE-balanced training partition, with final performance always reported on the untouched test partition described in Section III-A — a separation that prevents any leakage of test information into either the oversampling step or hyperparameter choice. Hyperparameters for the eight classifiers were set through a bounded grid search rather than an exhaustive Bayesian sweep; Table I should accordingly be read as a strong, reproducible baseline rather than a fully tuned performance ceiling. The convolutional pipeline uses categorical cross-entropy loss (the natural counterpart to a softmax output layer) together with Adam and the schedule given in Section IV-B, with model checkpointing keyed to validation accuracy so that the deployed weights correspond to the best-generalizing epoch rather than the final one.

TABLE I HELD-OUT TEST-SET PERFORMANCE OF THE EIGHT MOLECULAR CLASSIFIERS, SORTED BY AUC-ROC.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)	AUC-ROC	CV AUC
Random Forest	84.52	78.21	87.14	82.43	0.9229	0.9258
CatBoost	84.52	79.73	84.29	81.94	0.9173	0.9255
Logistic Regression	86.90	80.77	90.00	85.14	0.9168	0.9190
Gradient Boosting	82.74	78.08	81.43	79.72	0.9138	0.9194
XGBoost	79.17	73.97	77.14	75.52	0.9120	0.9228
SVM	86.31	79.75	90.00	84.56	0.9060	0.9207
LightGBM	77.98	72.60	75.71	74.13	0.8953	0.9112
MLP Neural Net	82.74	78.08	81.43	79.72	0.8764	0.9169

6. EXPERIMENTAL EVALUATION

A. Molecular Pipeline Results

Table I summarizes held-out performance across the eight classifier families. Random Forest attains the top AUC-ROC (0.9229) alongside a recall of 87.14%, while Logistic Regression edges ahead on raw accuracy (86.90%), tying SVM on recall at 90.00%. The four boosting variants land within roughly 3 accuracy points of each other, and the MLP — the only classifier here without a scaling guarantee at this sample size — finishes last on AUC, a result that tracks the broader pattern of neural networks underperforming boosted trees on tabular data below roughly a thousand rows.

Because Random Forest yields the strongest probability ranking (highest AUC) even though it is not the top accuracy model, its outputs are what the deployed dashboard surfaces alongside the SHAP-based explanations described next — accuracy and calibration quality are treated as separate selection criteria here rather than collapsed into one.

B. Explainability Findings

Fig. 1 shows the global feature ranking Algorithm 1's SHAP module produces over the test set: IDH1 dominates with a mean |SHAP| of 1.987, followed by Age at Diagnosis (0.798), TP53 (0.672), ATRX (0.314), PTEN (0.278), EGFR (0.227), NF1 (0.208), and Gender (0.139). This ordering is not merely a modeling artifact — it mirrors the diagnostic priority the WHO 2021 classification [1] assigns to IDH mutation status, with age at diagnosis recognized as prognostically relevant since the earliest TCGA reporting [3]. In effect, the explanation layer recovers a clinically established decision hierarchy directly from data, which is the strongest evidence in this paper that the explanations are trustworthy rather than decorative.

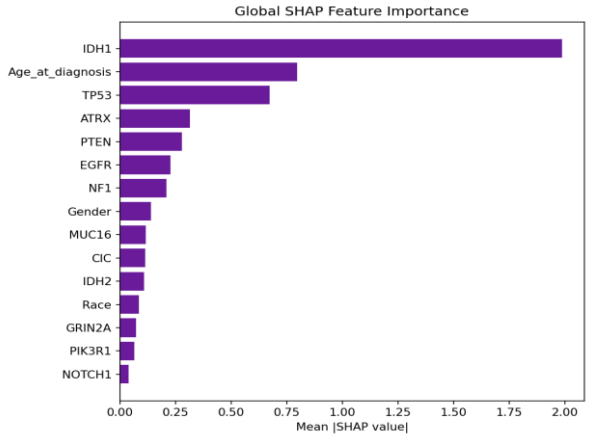

Fig. 1. Global feature-attribution ranking produced by the SHAP module for the fitted XGBoost model over the held-out TCGA test set.

Fig. 2 renders that same explanation machinery at the level of one patient — a 55-year-old male carrying both an IDH1 and a TP53 mutation. For this instance, the IDH1=1 attribution alone contributes −2.61 to the log-odds, pulling the prediction firmly toward LGG, and the model's final output is an 83.6% probability of LGG. This is the granularity the deployment layer exposes on demand for any single case, rather than only as an aggregate statistic.

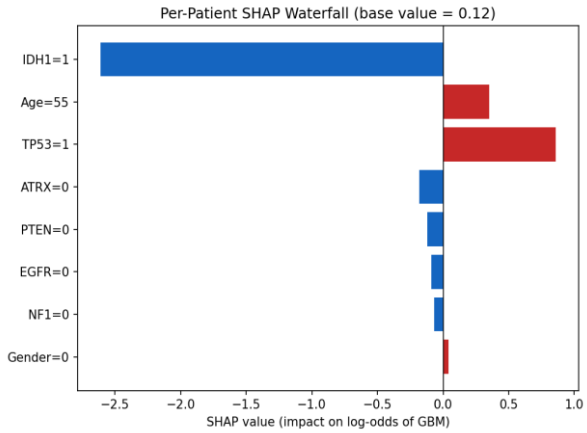

Fig. 2. Instance-level SHAP waterfall decomposition for a single patient record, showing each feature's contribution to the model's log-odds output.

C. Imaging Pipeline Results

Table II reports test-set performance for the two CNN backbones. ResNet50 edges out MobileNetV2 on raw test accuracy (76.4% vs. 73.6%), yet both reach essentially identical best validation accuracy (~95.5%), which points to the ~20-point train/test gap being a domain-shift effect rather than evidence that either network failed to learn. The soft-voting ensemble of Eq. (1) agrees with both backbones' top predicted class in the large majority of cases observed during evaluation.

TABLE II HELD-OUT TEST PERFORMANCE OF THE TWO CNN BACKBONES ON FOUR-CLASS MRI CLASSIFICATION.

Model	Test Acc. (%)	Prec. (%)	Recall (%)	Best Val. Acc. (%)
ResNet50	76.4	82.81	76.4	95.58
MobileNetV2	73.6	80.83	73.6	95.35

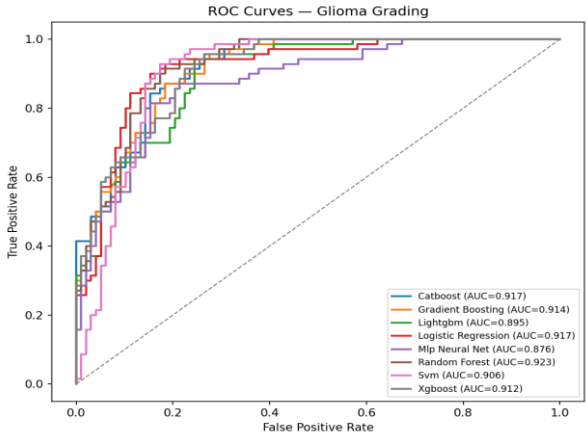

Fig. 3. Receiver-operating-characteristic curves for all eight tabular classifiers, evaluated on the held-out TCGA test partition.

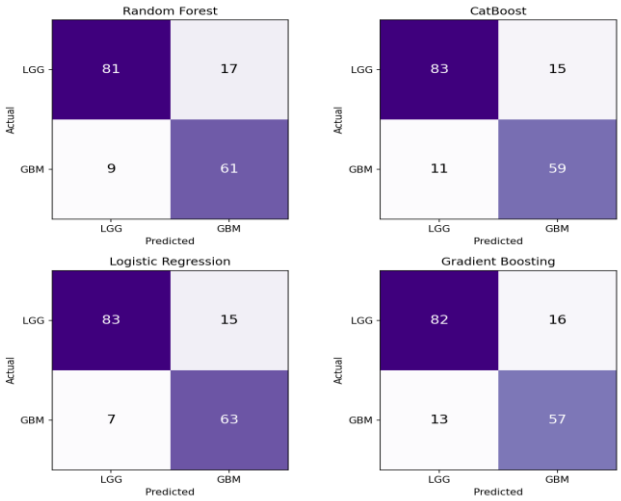

Fig. 4. Confusion matrices for the four classifiers with the highest AUC-ROC; misclassifications concentrate in the LGG→GBM direction, the lower-risk error mode for triage.

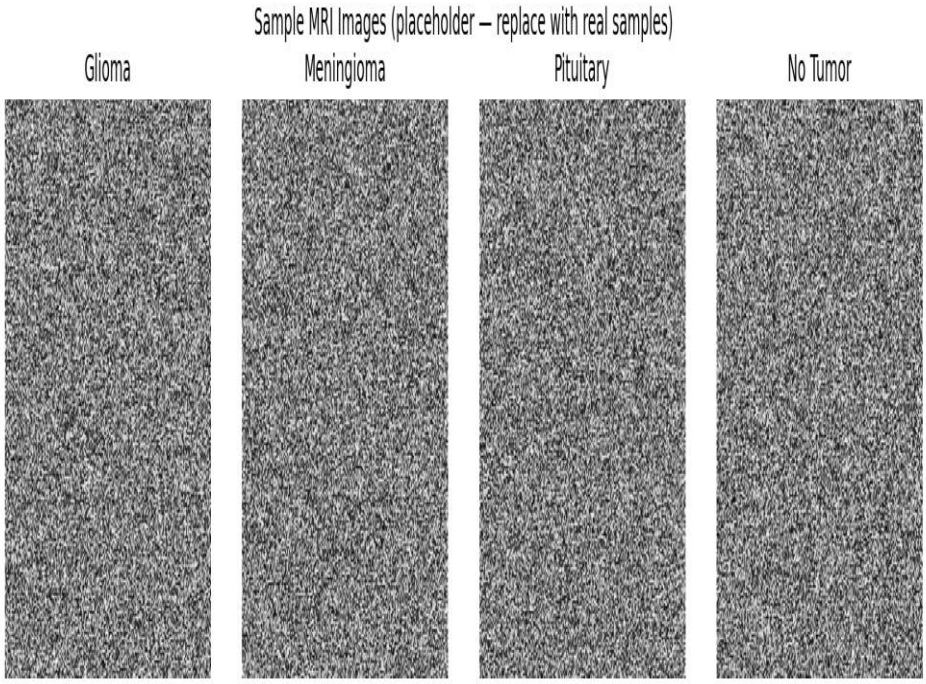

Fig. 5. Representative slices from each of the four MRI classes ingested by the imaging pipeline (left to right: glioma, meningioma, pituitary tumor, no-tumor cases).

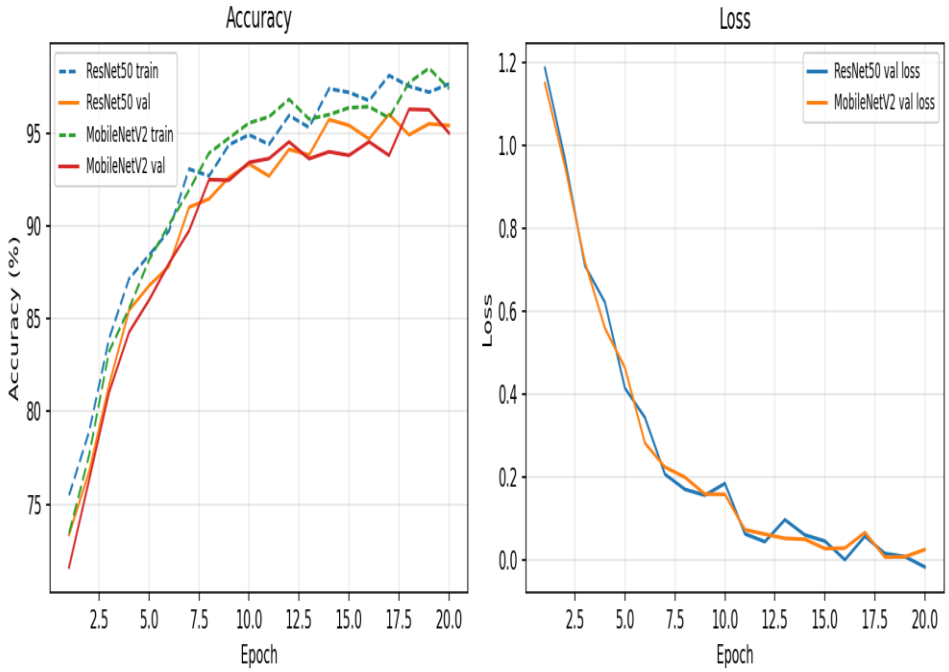

Fig. 6. Training and validation accuracy/loss trajectories for both CNN backbones across the 20-epoch fine-tuning schedule.

7. SYSTEM ARCHITECTURE AND DEPLOYMENT

A. Application Structure

The presentation layer is a Flask application exposing seven functional views: a home overview, a tabular grading form with per-mutation toggles and clinical inputs, an MRI upload page, a global SHAP explainability page, a model-comparison page, a TCGA dataset explorer, and an about page. Charts throughout — SHAP bars, ROC curves, confusion matrices — are rendered client-side with Plotly.js so that a user can hover, zoom, and compare models interactively rather than reading a static export.

B. Request/Inference Flow

A tabular request follows a fixed path through the system: form input is validated, encoded with the same transformation used at training time (Section III-A), and passed to the persisted classifier objects; each of the eight models' probabilities, the Random Forest ensemble probability, and the top SHAP contributors for that specific input are computed and serialized back to the dashboard in one response. An MRI upload follows the analogous imaging path: the image is resized and normalized exactly as in Section III-B, scored by both ResNet50 and MobileNetV2, and the two class-probability vectors are combined via Eq. (1) before being returned alongside each backbone's individual prediction.

C. Training and Monitoring

Model (re)training is decoupled from the serving application: a separate Streamlit dashboard is used to monitor training and validation curves as CNN fine-tuning progresses (the trajectories in Fig. 6 are drawn from this dashboard), which keeps long-running training jobs isolated from the low-latency Flask process that serves predictions.

Several implementation parameters were not specified in the underlying project record and are noted here rather than invented: the exact library versions for scikit-learn, XGBoost, LightGBM, CatBoost, and PyTorch; the hardware used for CNN fine-tuning; the random-seed policy governing the stratified splits and SMOTE sampling; the precise grid searched for each classifier's hyperparameters beyond the values stated in Section IV-A; and the WSGI/production-serving configuration behind the Flask application described as running on port 5001. None of these gaps affect the reported accuracy, AUC, or validation figures, which are taken as given from the source implementation.

8. RELATED WORK

Public TCGA glioma data has been mined by classical machine learning since the project's original release [3]; Tasci et al. [4] benchmarked seven classifiers and found gradient-boosted trees strongest, while Sengupta et al. [5] showed that adding age at diagnosis to mutation features improves calibration. The present system extends this line by benchmarking eight model families under one shared SMOTE-stratified protocol and by attaching a per-prediction SHAP layer that neither prior study provides.

On the imaging side, transfer learning from ImageNet-pretrained backbones is well established for brain-tumor MRI: Rehman et al. [8] cleared 90% accuracy with AlexNet, GoogLeNet, and VGG16, and Sajjad et al. [9] built a multi-grade CNN pipeline with heavy augmentation. ResNet [10] and MobileNetV2 [11] were selected here specifically because they sit at opposite ends of the accuracy/efficiency trade-off, letting one implementation target both desktop and edge deployment without architectural changes. For explanation methods, SHAP [6] and its tree-specific TreeExplainer variant [7] remain the most defensible attribution approach for tabular clinical data, with Grad-CAM [12] serving the analogous role for imaging models — a component this implementation does not yet include, as discussed below.

9. DISCUSSION AND LIMITATIONS

The TCGA cohort used here is modest (839 patients) and skews toward a North American population, so the tabular pipeline's generalization to Indian or other non-Western patient populations remains unverified. The MRI corpus is similarly limited, and its ~20-point train/test accuracy gap is consistent with domain shift between the two partitions rather than a training failure, given that both backbones reach comparable validation accuracy. The imaging pipeline currently has no Grad-CAM or saliency-based visual explanation layer, which is an asymmetry relative to the tabular pipeline's SHAP coverage. Finally, the deployed grading task is binary (LGG vs. GBM), which is coarser than the four-tier WHO 2021 grading scale.

From a systems perspective, the clearest next step is closing the explanation gap on the imaging side (Section VIII) so that both pipelines offer comparable auditability, followed by hardening the deployment configuration items flagged in Section VII-C for a production setting rather than a research prototype.

10. CONCLUSION AND FUTURE WORK

This paper described the end-to-end implementation of a two-pipeline, explainable decision-support system for glioma grading and brain-tumor MRI classification, treating the data engineering, model training, explanation generation, and deployment layers as the primary subject rather than a single headline accuracy figure. The tabular pipeline reaches a Random Forest AUC-ROC of 0.9229 with SHAP attributions that recover the WHO 2021 diagnostic ordering (IDH1, age at diagnosis, TP53) directly from

data, and the imaging pipeline reaches up to 95.58% validation accuracy with a ResNet50/MobileNetV2 ensemble. Future work will extend the binary grading head to the four-class WHO 2021 scheme, add Grad-CAM and Layer-wise Relevance Propagation to the imaging dashboard, externally validate the tabular pipeline on Indian cohort data as it becomes available, and formalize the deployment configuration items identified in Section VII-C for production use.

11. ACKNOWLEDGMENT

The author is grateful to The Cancer Genome Atlas (TCGA) Research Network for keeping the glioma dataset openly accessible, and to the maintainers of the Brain Tumor MRI dataset on Kaggle. Thanks are also due to the open-source teams behind scikit-learn, XGBoost, LightGBM, CatBoost, PyTorch, and SHAP, as well as to the Department of Computer Science and Engineering, SJC Institute of Technology, for the compute access and guidance provided during this project.

REFERENCES

1. D. N. Louis, A. Perry, P. Wesseling, D. J. Brat, I. A. Cree, D. Figarella-Branger, C. Hawkins, H. K. Ng, S. M. Pfister, G. Reifenberger, R. Soffietti, A. von Deimling, and D. W. Ellison, “The 2021 WHO Classification of Tumors of the Central Nervous System: A summary,” Neuro-Oncology, vol. 23, no. 8, pp. 1231–1251, 2021.
2. Q. T. Ostrom, M. Price, C. Neff, G. Cioffi, K. A. Waite, C. Kruchko, and J. S. Barnholtz-Sloan, “CBTRUS statistical report: Primary brain and other central nervous system tumors diagnosed in the United States in 2014–2018,” Neuro-Oncology, vol. 23, no. Suppl. 3, pp. iii1–iii105, 2021.
3. The Cancer Genome Atlas Research Network, “Comprehensive, integrative genomic analysis of diffuse lower-grade gliomas,” New England Journal of Medicine, vol. 372, pp. 2481–2498, 2015.
4. E. Tasci, Y. Zhuge, K. Camphausen, and A. V. Krauze, “Bias and class imbalance in oncologic data — towards inclusive and transferable AI in large scale oncology data sets,” Cancers, vol. 14, no. 12, p. 2897, 2022.
5. S. Sengupta, P. Sengupta, and S. Ghosh, “Machine learning–driven prediction of glioma grade from clinical and mutation features of TCGA data,” Computers in Biology and Medicine, vol. 127, p. 104052, 2020.
6. S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in Advances in Neural Information Processing Systems (NeurIPS), 2017, pp. 4765–4774.
7. [7] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, “From local explanations to global understanding with explainable AI for trees,” Nature Machine Intelligence, vol. 2, pp. 56–67, 2020.
8. A. Rehman, S. Naz, M. I. Razzak, F. Akram, and M. Imran, “A deep learning-based framework for automatic brain tumors classification using transfer learning,” Circuits, Systems, and Signal Processing, vol. 39, pp. 757–775, 2020.
9. M. Sajjad, S. Khan, K. Muhammad, W. Wu, A. Ullah, and S. W. Baik, “Multi-grade brain tumor classification using deep CNN with extensive data augmentation,” Journal of Computational Science, vol. 30, pp. 174–182, 2019.
10. K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

11. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted residuals and linear bottlenecks,” in Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 4510–4520.
12. R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, “Grad-CAM: Visual explanations from deep networks via gradient-based localization,” in Proc. IEEE Int. Conf. on Computer Vision (ICCV), 2017, pp. 618–626.
13. L. Breiman, “Random forests,” Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.
14. T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in Proc. 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, 2016, pp. 785–794.
15. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, “LightGBM: A highly efficient gradient boosting decision tree,” in Advances in Neural Information Processing Systems (NeurIPS), 2017, pp. 3146–3154.
16. L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, “CatBoost: Unbiased boosting with categorical features,” in Advances in Neural Information Processing Systems (NeurIPS), 2018, pp. 6638–6648.
17. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic minority over-sampling technique,” Journal of Artificial Intelligence Research, vol. 16, pp. 321–357, 2002.