Explainable AI on the AIFB RDF Dataset (Strategy 2)

Bobby Patel, Mohit Jain, and Tanmay Mhatre

University of Paderborn, Germany
{bpatel, jmohit, tmhatre}@mail.upb.de

Abstract. This project explores explainable machine learning on RDF graph-structured data using the AIFB dataset. The RDF data is transformed into a tabular format by extracting literal predicates as features. A Random Forest classifier is trained to predict research group affiliations, and model explanations are provided using SHAP (Shapley Additive Explanations). We demonstrate both global and local interpretability techniques and evaluate the performance and explainability trade-offs of classical ML models applied to knowledge graph data.

1 Introduction

Knowledge graphs have emerged as a foundational structure for representing rich semantic data. The Resource Description Framework (RDF) enables the encoding of entities and relationships using subject-predicate-object triples, and has been widely adopted in domains such as the Semantic Web, bioinformatics, and academic knowledge bases. Despite their expressive power, RDF graphs present challenges for conventional machine learning (ML) algorithms due to their inherently non-tabular nature.

Explainable AI (XAI) addresses the need for transparency in AI systems, especially when models are deployed in sensitive applications such as healthcare or research evaluation. Understanding not just *what* the model predicts but also *why* it makes those predictions is essential for trust and accountability. This project bridges the gap between RDF-based data and classical ML interpretability tools.

In this mini-project, we adopt a strategy that first flattens RDF data into a tabular structure by extracting literal features, which are then used to train a Random Forest classifier. We then use SHAP (Shapley Additive Explanations) to gain insight into both global and local decision processes of the model. Our results show that even simple models like Random Forests, when paired with XAI techniques, can provide transparent and reasonably accurate predictions on RDF datasets.

2 Data Analysis

2.1 Dataset Overview

The AIFB RDF dataset captures metadata related to researchers, their publications, affiliations, and contact details. It is structured as RDF triples, making it

a suitable candidate for both graph-based machine learning (ML) and symbolic reasoning tasks. The dataset contains:

- 8,277 RDF triples
- 828 unique subjects (e.g., persons, organizations)
- 45 unique predicates, including name, email, phone, and title
- Only literal predicates were used for modeling, to simplify interpretation

2.2 RDF to Tabular Conversion

Using the `rdflib` Python package, we extracted only those triples where the object is a literal (e.g., strings or numbers). Each person in the dataset was mapped to a row, with each literal predicate becoming a column. Missing values were filled with NaN, and categorical features were one-hot encoded. This step enabled compatibility with traditional machine learning models such as Random Forests.

2.3 Train-Test Split

Training and test splits were provided as TSV files:

- **trainingSet.tsv:** 160 labeled training instances
- **testSet.tsv:** 36 labeled testing instances

Target labels refer to research group URIs, which were mapped to readable names during post-processing.

3 Model Training and Evaluation

3.1 Preprocessing

The feature matrix underwent several preprocessing steps to ensure compatibility with the learning algorithm:

- Categorical features were one-hot encoded using Scikit-learn's `OneHotEncoder`, applied separately to the training data to avoid data leakage.
- Missing values introduced by sparse predicates were filled with zeros.
- Feature scaling was not necessary for tree-based models such as Random Forest.

3.2 Classifier Description

We selected the Random Forest Classifier from Scikit-learn (`sklearn.ensemble.RandomForestClassifier`) for its interpretability, robustness, and native support for categorical and sparse data. Default hyperparameters were used in this baseline setup. The ensemble nature of the classifier helps mitigate overfitting while retaining sensitivity to feature interactions, which is important for semantic datasets.

3.3 Evaluation Metrics and Observations

The classifier was evaluated using precision, recall, F1-score, and support per class on the test set. The model performs well on majority classes (e.g., Business Info & Communication Systems) but shows weak generalization on minority groups, a common issue in imbalanced datasets. This underscores the importance of class-aware evaluation when applying machine learning to real-world, heterogeneous data.

3.4 Results

Table 1. Performance metrics for the Random Forest Classifier on the AIFB dataset.

Class	Precision	Recall	F1-score	Support
`id1instance`	1.00	0.87	0.93	15
`id2instance`	1.00	0.17	0.29	6
`id3instance`	0.60	1.00	0.75	12
`id4instance`	0.50	0.33	0.40	3
Accuracy			**0.75**	36
Macro avg	0.78	0.59	0.59	36
Weighted avg	0.82	0.75	0.72	36

- The model predicts the majority class (`id1instance`) with both perfect precision and the highest recall among all classes, indicating that almost all samples from this group are correctly identified.
- Despite achieving perfect precision for `id2instance`, the recall for this class is very low (0.17), which means the model fails to identify most true positives for this class, highlighting a substantial class imbalance effect.
- The class `id3instance` stands out with perfect recall (1.00), meaning all actual instances are detected, but with a moderate precision (0.60), so there are more false positives for this group.
- For the smallest class (`id4instance`), both precision (0.50) and recall (0.33) are limited, reflecting the model's difficulty with rare classes and its tendency to misclassify these cases.
- Macro-averaged metrics reveal a clear performance drop for minority classes, as seen by the lower macro recall and F1-score (0.59), compared to the weighted averages, which are dominated by the majority class performance.

4 Model Explanation

4.1 Global SHAP Summary Plots

The SHAP summary plots below display the global importance of each feature for every class. These plots illustrate how different features influenced predictions across the dataset for each research group (class).

Note on reproducibility: Due to the stochastic nature of the Random Forest model, SHAP force plots—which explain individual instance predictions—may vary slightly each time the model is retrained. This is because force plots are sensitive to the exact decision paths taken for a single instance. However, the SHAP summary plots remain largely consistent, as they reflect the average feature importance over the entire dataset. In our experiments, the evaluation metrics (accuracy, F1-score, precision, recall) and test set predictions remained stable because a fixed random state was used during training.

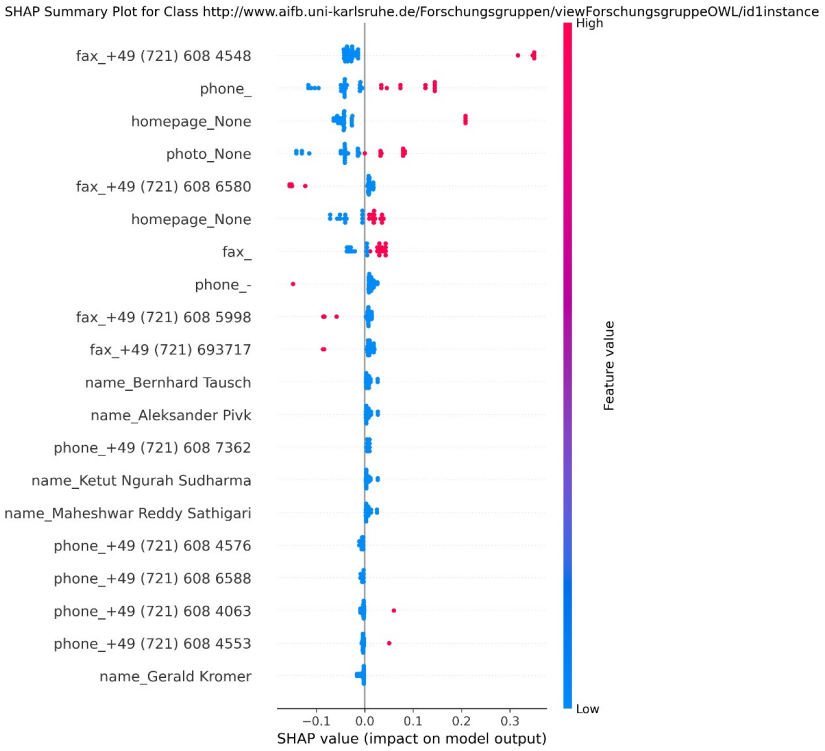

Fig. 1. SHAP summary plot for Class 0.

Key points for Class 0:

- `fax_+49 (721) 608 4548` is the most influential feature for this class label; its presence (red) pushes instances strongly towards this class.
- The absence of `fax_+49 (721) 608 6580` (blue dots) also pushes certain instances towards this class.
- Features such as `phone_`, `homepage_None`, and `photo_None` have significant, though sometimes moderate, effects.
- In the summary plot, red points indicate that a high (present) value of the feature increases the probability of this class, while blue points (feature absent) have the opposite effect.
- Features with SHAP values to the right of 0.0 push the prediction toward this class; features to the left push away.
- Each dot represents a single instance, with feature importance ranked by mean absolute SHAP value.

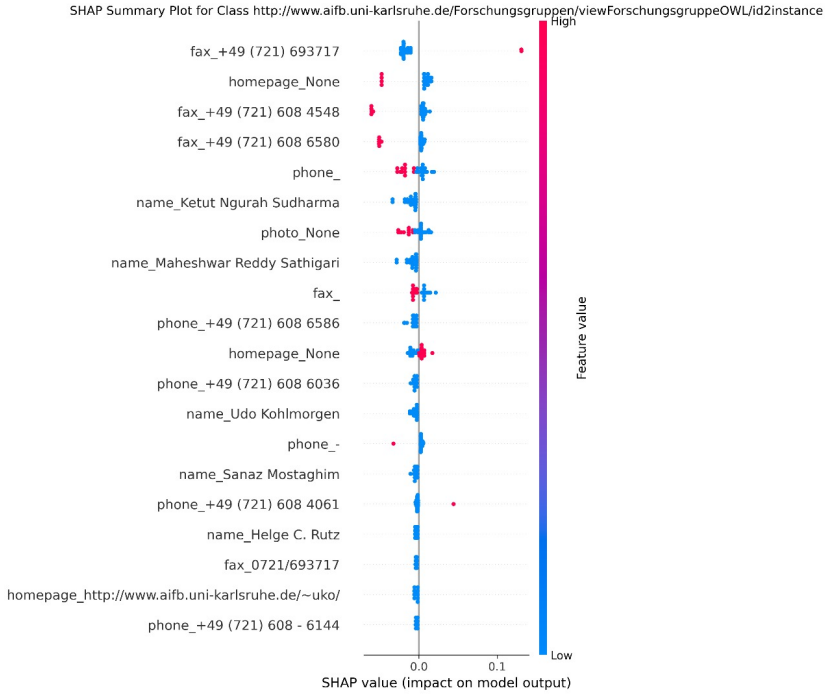

Fig. 2. SHAP summary plot for Class 1.

Key points for Class 1:

- `fax_+49 (721) 693717` is the most influential; its presence (red) pushes predictions toward this class.
- `homepage_None` and `fax_+49 (721) 608 4548` are the second and third most important features, respectively.

– Some features, when absent (blue), push instances towards the class; in other cases, presence pushes away, showing nuanced effects of missing or present data.
– Features with SHAP values of 0 have no impact on predictions for this class.
– All features are one-hot encoded, so color reflects presence (red) or absence (blue).
– SHAP values to the right of 0.0 indicate features pushing toward the class; left means pushing away.

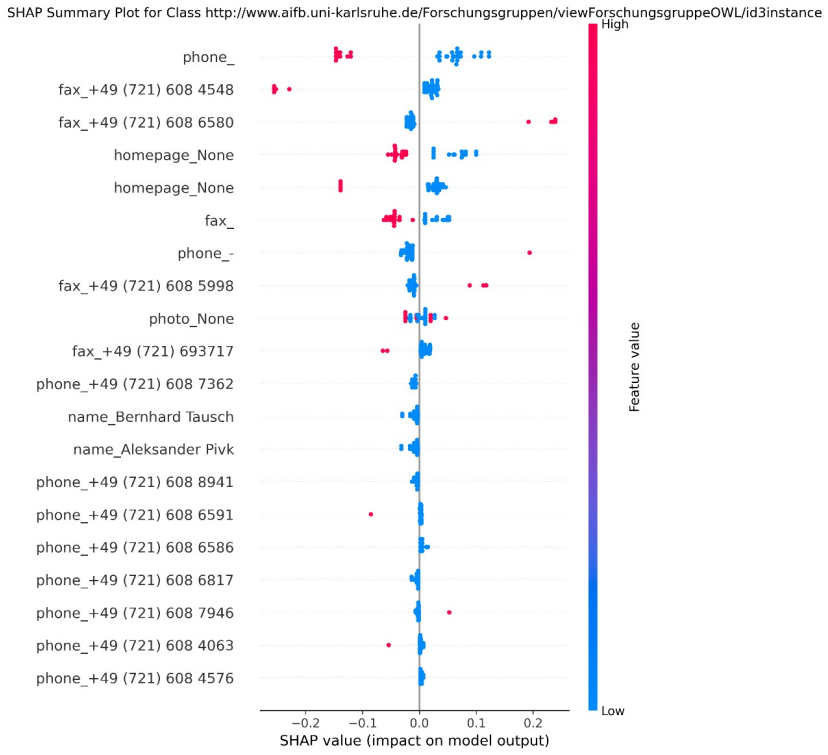

Fig. 3. SHAP summary plot for Class 2.

Key points for Class 2:

– `phone_` is the most influential; its absence (blue dots) actually pushes many instances toward this class.
– `fax_+49 (721) 608 4548` and `fax_+49 (721) 608 6850` are also highly influential for this class label.
– The feature `photo_None` is unique in that both its presence and absence can push different instances toward this class, illustrating complex decision boundaries.

– Features to the right of 0.0 increase the likelihood of class 2; features to the left decrease it.
– Each dot corresponds to a single sample, and colors indicate feature presence (red) or absence (blue).

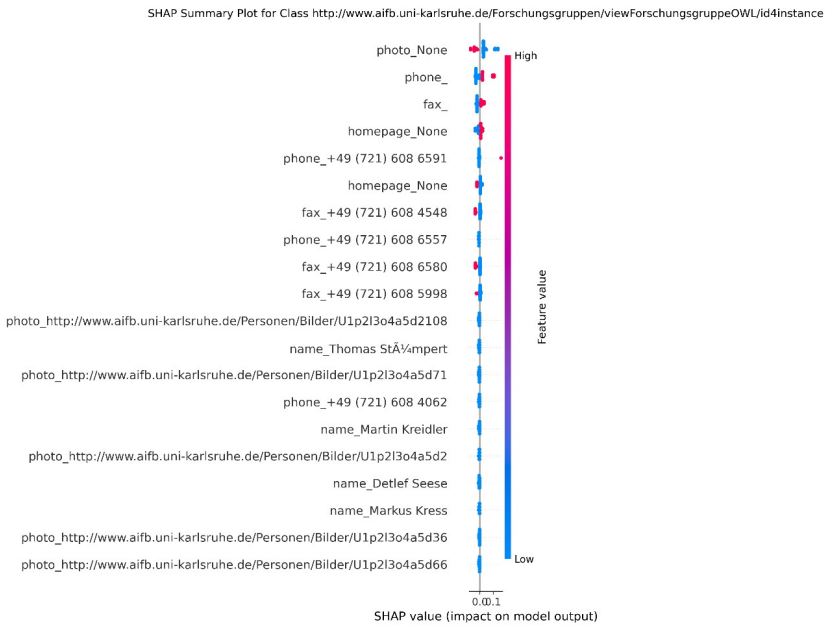

Fig. 4. SHAP summary plot for Class 3.

Key points for Class 3:

– `photo_None` is the most influential feature for this class.
– The absence of `photo_None` (blue) pushes predictions toward this class, while its presence (red) pushes them away.
– Conversely, the presence of `fax_` and `phone_` pushes instances toward this class, and their absence pushes away.
– This summary plot also demonstrates that the same feature may have different effects in different classes.
– Features to the right of 0.0 push predictions toward class 3; to the left, away.

Table 2. Top 20 Features Ranked by Mean Absolute SHAP Value

Rank	Feature	Mean |SHAP value|
1	phone_	0.049180
2	fax_+49 (721) 608 4548	0.036904
3	homepage_None	0.032079
4	photo_None	0.030642
5	fax_+49 (721) 608 6580	0.025471
6	homepage_None	0.024095
7	fax_	0.023379
8	fax_+49 (721) 693717	0.013353
9	phone_-	0.012020
10	fax_+49 (721) 608 5998	0.011627
11	phone_+49 (721) 608 7362	0.004929
12	phone_+49 (721) 608 6591	0.004907
13	name_Ketut Ngurah Sudharma	0.004450
14	name_Bernhard Tausch	0.004018
15	name_Aleksander Pivk	0.003964
16	name_Maheshwar Reddy Sathigari	0.003718
17	phone_+49 (721) 608 6586	0.003507
18	phone_+49 (721) 608 6557	0.003342
19	phone_+49 (721) 608 4576	0.002656
20	phone_+49 (721) 608 8941	0.002527

Key points for Top 20 Features:

- The feature `phone_` has the highest mean absolute SHAP value, indicating that the presence or absence of general phone information is the most influential factor across model predictions.
- Fax-related features, such as `fax_+49 (721) 608 4548` and `fax_+49 (721) 608 6580`, are also highly predictive for certain research groups.
- The feature `homepage_None` ranks third and sixth, showing that missing homepage information has a substantial impact on classification.
- The presence or absence of personal attributes such as `photo_None` and specific names demonstrates the value of both unique identifiers and missing data.
- Overall, contact information (phones, faxes, homepages) dominates the top-ranked features, underlining the importance of such metadata in distinguishing research group affiliations in the AIFB dataset.

4.2 Local SHAP Force Plots

To understand the model's behavior for individual predictions, we use SHAP force plots. These visualizations illustrate how each feature contributed to a specific prediction by pushing the model output above or below the expected base value.

Interpretation:

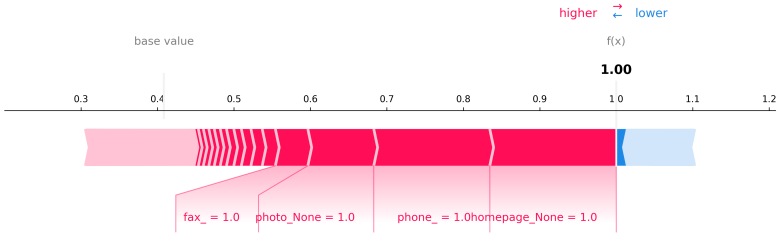

Fig. 5. SHAP force plot for a perfectly classified instance (predicted $f(x) = 1.00$). Features like `fax_`, `photo_None`, `phone_`, and `homepage_None` are all present and strongly pushed the prediction toward the correct class.

- The model is highly confident, with $f(x) = 1.00$ far to the right of the base value.
- All visible features are present (value = 1) and colored red, indicating they substantially increased the predicted probability for this class.
- Each feature's strong positive contribution is clear from the length of the red arrows.
- Such cases demonstrate how strong feature evidence leads to correct and confident predictions.

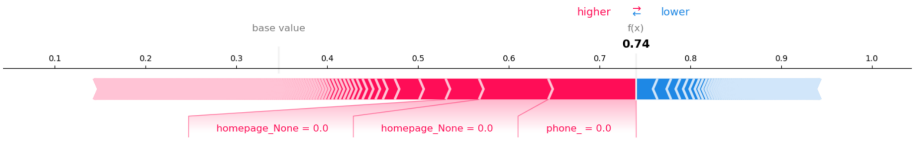

Fig. 6. SHAP force plot for a correctly classified instance (predicted $f(x) = 0.74$ for class 2). Features like `homepage_None` and `phone_` are absent (value = 0.0) but still contribute positively, as shown by the red arrows.

Interpretation:

- The model predicts class 2 with relatively high confidence ($f(x) = 0.74$), with the base value near 0.3–0.4.
- Absence of features (value = 0.0) in this context still pushes the prediction higher, indicating the model has learned that missing these features is informative for this class.
- All contributing arrows are red and positioned to the left of $f(x)$, showing a positive impact on the output.

– Such behavior reflects how, for some classes, the absence of metadata can be a strong positive signal.

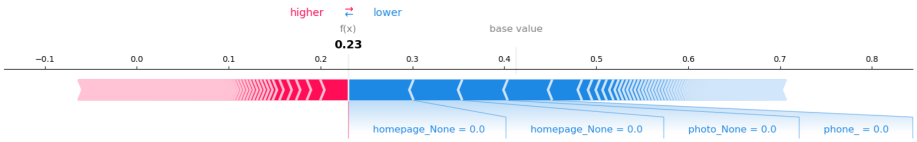

Fig. 7. SHAP force plot for an incorrectly classified instance (model output $f(x) = 0.23$ for class 0). All visible features are absent (value = 0.0), and their blue arrows show a small positive contribution, but not enough to change the prediction.

Interpretation:

– The base value is around 0.4–0.5, but the SHAP contributions lower the model output to $f(x) = 0.23$.
– All features (`homepage_None`, `photo_None`, `phone_`) are absent (value = 0.0) and contribute slightly, as shown by blue arrows to the right of $f(x)$.
– The combined effect of these low values increases the probability for this class a little, but not enough to overcome the model's overall confidence in another class.
– This example shows how weak feature signals can result in misclassification when the evidence is insufficient.

These force plots highlight how both the presence and absence of features can influence individual predictions, and how the model combines these signals for each instance. They provide transparent, instance-level explanations, showing why the model is confident, uncertain, or wrong in different cases.

5 Conclusion

This study demonstrated that converting RDF graph data into a tabular format enables the use of classical machine learning models, such as Random Forests, for research group classification on the AIFB dataset. By applying SHAP, we were able to interpret both global and local model predictions and identify which features most influenced each decision.

Our analysis showed that the model performed well on majority classes but struggled with minority and imbalanced classes, as revealed by lower recall and F1-scores for these groups. The SHAP explanations highlighted the importance of contact-related features—such as phone, fax, and homepage attributes—and also showed how both the presence and absence of specific metadata can impact predictions.

Although the Random Forest classifier achieved stable evaluation metrics with a fixed random state, local SHAP force plots illustrated the potential variability in individual predictions upon retraining. These findings underline the importance of model transparency, especially when class distributions are skewed and predictions must be trusted.

6 Contributions of Team Members

- Bobby Patel: Data loading, RDF parsing, SHAP visualizations, documentation.
- Mohit Jain: Feature engineering, model training, hyperparameter tuning, documentation.
- Tanmay Mhatre: Local explanation analysis, documentation, report integration.

References

1. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., Bizer, C.: AIFB Dataset. Figshare. https://figshare.com/articles/dataset/AIFB_DataSet/745364/1
2. Heindorf, S.: Explainable AI course material, University of Paderborn (2025)
3. Lundberg, S. M., & Lee, S.-I.: A unified approach to interpreting model predictions. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS), pp. 4765–4774, 2017.
4. Lundberg, S. M., Erion, G., & Lee, S.-I.: Consistent individualized feature attribution for tree ensembles. arXiv preprint arXiv:1802.03888, 2018.
5. Breiman, L.: Random forests. Machine Learning, 45(1):5–32, 2001.
6. Molnar, C.: Interpretable Machine Learning. Online Book. https://christophm.github.io/interpretable-ml-book/, 2022. (Accessed July 2025).
7. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12:2825–2830, 2011.
8. McKinney, W.: Data structures for statistical computing in Python. In Proceedings of the 9th Python in Science Conference (SciPy), pp. 51–56, 2010.
9. Berners-Lee, T., Hendler, J., & Lassila, O.: The Semantic Web. Scientific American, 284(5):34–43, 2001.
10. Berrueta, D., & Álvarez, J.: RDF: Resource Description Framework. W3C Tutorial, 2004.