mGPS Algorithm Optimization

Course: Bioinformatics Research Project (BINP37), 15 credits

Student: Chandrashekar CR

(email: ch1131ch-s@student.lu.se)

Supervisor: Eran Elhaik

(email: eran.elhaik@biol.lu.se)

Lund University 2025

1. Results

1.1 Overview

This section presents the performance evaluation of various hierarchical machine learning models for geographic prediction using m etagenomic data. We compare the effectiveness of separate neural networks, combined neural networks, GrowNet, and ensemble learning approaches in predicting geographic origins at continent, city, and coordinate levels.

1.2 Dataset and Evaluation Metrics

We evaluated all models on the filtered MetaSUB dataset, containing 4,070 samples from 40 cities on 7 continents. Data were partitioned into training, validation, and test sets (2,604/652/814 samples, respectively) after quality control. The dataset exhibits class imbalance, particularly at the continent and city levels.

Principal metrics of evaluation are **classification accuracy**, **macro-averaged F1-score**, and **weighted F1-score** for categorical predictions at both continent and city scales. For geospatial accuracy estimation, we measured **geodesic error**, the great-circle distance between predicted and actual coordinates on Earth's surface. **??** We also provide **in-radius accuracy** (the proportion of predictions within specified geodesic distances of the true location). On classification tasks, **AUPR** (area under the precision-recall curve) and **AUC** (area under the ROC curve) are only reported for the ensemble model to facilitate a balanced comparison with the mGPS state-of-the-art model. (Zhang et al., 2024)

1.3 Evaluation Metrics Explanation

Accuracy is the quantity of correct predictions compared to all samples. **Macro-averaged F1-score** calculates the F1-score for each class independently, and then averages these F1-scores, treating all classes equally. **Weighted F1-score** also calculates F1-score for each class independently, and averages them using a weighting of the number of true instances per class. This will make the metrics more robust to class imbalance. Metrics are reported both at the continent and city level.

Geodesic error is the great-circle distance (km) between the predicted and true coordinates on the surface of the Earth; this is the most direct measure of spatial prediction accuracy. **In-radius accuracy** is the proportion of predictions within a predetermined geodesic distance from the true location (for example within 50 km, 100 km, etc.).

In the case of the coordinate regression, we also report **RMSE** (Root Mean Square Error), the square root of the average squared distance between predicted and true coordinates; **MAE** (Mean Absolute Error), the average of the absolute distances; and R^2

(coefficient of determination), which is the proportion of variation in the true coordinates explained by the model. This comprehensive set of metrics allows proper evaluation of hierarchical geographic prediction performance. **AUC** (Area Under the ROC Curve) measures the ability of the model to distinguish between classes, summarizing the trade-off between true positive rate and false positive rate across thresholds. **AUPR** (Area Under the Precision-Recall Curve) evaluates the trade-off between precision and recall, which is especially informative for imbalanced datasets. Both metrics provide insight into classification performance beyond simple accuracy.

1.4 Model Performance

This section presents the performance of the various models evaluated on the MetaSUB dataset, focusing on continent and city classification accuracy, geodesic error, and in-radius accuracy. The results are summarized in Table 1.

Table 1. Comparison of model performance across continent and city metrics, and error group analysis.

Model	Acc.	Avg F1	Wtd F1	Acc.	Avg F1	Wtd F1	Mean	Median	Prop.	Wtd	Mean	Median	Prop.	Wtd	Mean	Median	Prop.	Wtd	Mean	Median	Prop.	Wtd
Separate NN	0.85	0.78	0.85	0.70	0.55	0.71	3994	3255	0.694	2772	5333	3703	0.155	826	7668	8555	0.007	57	9098	7532	0.144	1308
Combined NN	0.83	0.75	0.83	0.75	0.45	0.72	502	274	0.714	358	2101	1523	0.113	237	3434	2252	0.036	122	6637	5377	0.138	913
GrowNet	0.86	0.77	0.86	0.75	0.60	0.76	904	599	0.742	671	2215	1710	0.122	269	4501	4324	0.009	39	7090	5896	0.128	906
Ensemble	0.95	0.89	0.95	0.93	0.80	0.92	208.1	12.3	0.903	187.9	2148.1	1713.5	0.045	97.6	3902.2	3534.2	0.022	86.3	7365.5	6822.9	0.029	217.2

Notes: Acc. = Accuracy; Avg F1 = Macro-averaged F1 score; Wtd F1 = Weighted F1 score.

Error group columns: Cc-Zc = Continent correct, City correct; Cc-Zi = Continent correct, City incorrect; Ci-Zc = Continent incorrect, City correct; Ci-Zi = Continent incorrect, City incorrect.
For each group: Mean/Median Error (km), Proportion of samples, and Weighted Error.

1.4.1 Separate Neural Networks

The separate neural network approach was evaluated in three sequential stages: continent classification, city classification, and coordinate regression.

Continent Classification The continent classifier achieved a test accuracy of 84.9% with a macro-averaged F1-score of 0.78 and a weighted F1-score of 0.85, indicating decent performance across continents despite class imbalance. Supplementary Table 22 presents detailed classification metrics.

City Classification The city classifier achieved a test accuracy of 70.1%, a macro-averaged F1-score of 0.55, and a weighted F1-score of 0.71. The lower macro-averaged F1-score compared to weighted F1-score reflects the effect of class imbalance, with underrepresented cities showing lower classification performance. Supplementary Table 26 presents a detailed city classification metrics.

Coordinate Regression The coordinate regression model achieved an RMSE (Root Mean Square Error) of 0.581, MAE (Mean Absolute Error) of 0.276, and coefficient of determination (R^2) of 0.658 on the test set. Geodesic error analysis revealed a median error of 4,237 km, mean error of 4,962 km, and maximum error of 17,788 km. Supplementary Table 30 presents a detailed error breakdown by prediction correctness.

In-radius accuracy analysis revealed that only 1.8% of predictions were within 1,000 km of the true location, while 55.7% were within 5,000 km (Supplementary Table 33). These metrics indicate that the separate neural networks approach, while providing reasonable classification performance, struggles with precise coordinate prediction.

1.4.2 Combined Neural Networks

The combined hierarchical neural network jointly predicts continent, city, and coordinates using a unified architecture with weighted multi-task learning. On the test set, this model achieved 82.7% continent a ccuracy (macro F1-score: 0.75, weighted F1-score: 0.83; Supplementary Table 23) and 74.9% city accuracy (macro F1-score: 0.45, weighted F1-score: 0.72; Supplementary Table 27). For coordinate regression, the model achieved an RMSE of 0.237, MAE of 0.126, and R^2 of 0.699. The median geodesic error decreased substantially to 519 km, with a mean error of 1,631 km and maximum error of 19,604 km. Supplementary Table 31 provides a detailed error analysis by prediction group. In-radius accuracy showed marked improvement, with 66.3% of predictions within 1,000 km and 89.3% within 5,000 km (Supplementary Table 33). These results demonstrate that the combined neural network approach significantly outperforms separate networks for coordinate prediction while maintaining comparable classification performance.

1.4.3 Hierarchical GrowNet

GrowNet, which combines neural networks with gradient boosting principles (Feng et al., 2021), achieved the highest classification accuracy among neural models. It reached 86.4% continent accuracy (macro F1-score: 0.77, weighted F1-score: 0.86; Supplementary Table 24) and 75.1% city accuracy (macro F1-score: 0.60, weighted F1-score: 0.76; Supplementary Table 28).

For coordinate regression, GrowNet achieved a median geodesic error of 823 km and mean error of 1,885 km, with a maximum error of 18,964 km. The coordinate regression MSE was 0.318, RMSE was 0.558, and R^2 was 0.685. The in-radius accuracy was 57.4% within 1,000 km and 89.1% within 5,000 km (Supplementary Table 33). Supplementary Table 32 provides a detailed error analysis by prediction group. Compared to both separate and combined neural networks, GrowNet showed lesser performance in city prediction accuracy to the combined neural network approach.

1.4.4 Ensemble Learning Model

Our ensemble learning approach, which integrates multiple models , achieved state-of-the-art results across all prediction tasks. This superior performance aligns with empirical findings that ensemble methods often outperform individual models (Opitz and Maclin, 1999; Mahdavi-Shahri et al., 2016). The ensemble attained 95.0% continent accuracy (macro F1-score: 0.89, weighted F1-score: 0.95; Supplementary Table 25) and 93.0% city accuracy (macro F1-score: 0.80, weighted F1-score: 0.92; Supplementary Table 29), with TabPFN delivering exceptional coordinate regression performance.

Continent Classification The ensemble model achieved the highest continent classification accuracy (95.0%) among all approaches. Even for underrepresented continents like Oceania, the model maintained reasonable performance, with a macro-averaged F1-score of 0.89 and weighted F1-score of 0.95 across all continents (Supplementary Table 25).

City Classification City classification proved similarly successful, with both XGBoost and LightGBM exceeding 91% accuracy in cross-validation. The final meta-model achieved a test accuracy of 93%, macro F1-score of 0.80, and weighted F1-score of 0.92, representing a substantial improvement over all neural approaches (Supplementary Table 29). This high accuracy at both continent and city levels provides a strong foundation for accurate coordinate prediction.

Coordinate Regression and Geodesic Error For coordinate regression, the ensemble leveraged TabPFN, which achieved exceptional geospatial precision. The test set median distance error was just 13.72 km, with a mean distance error of 589.02 km and a 95th percentile error of 3,577.48 km. Table 2 provides a detailed analysis of error distribution across prediction groups.

Table 2. Ensemble Learning Model: Error Group Analysis

Group	Count	Mean Error (km)	Median Error (km)	Proportion	Weighted Error
C_correct Z_correct	735	208.13	12.33	0.9029	187.93
C_correct Z_wrong	37	2148.09	1713.46	0.0455	97.64
C_wrong Z_correct	18	3902.22	3534.17	0.0221	86.29
C_wrong Z_wrong	24	7365.53	6822.91	0.0295	217.17

Note: C = Continent, Z = City. Groups indicate correctness of continent and city predictions.

When both continent and city predictions are correct (90.3% of cases), the median error drops dramatically to just 12.3 km.

The distribution of geodesic errors by continent and city (Figure 1) shows that most predictions fall within small distance bins, especially for well-represented regions (Supple-

mentary Table 25). This highlights the model's ability to achieve high spatial precision for the majority of test samples.

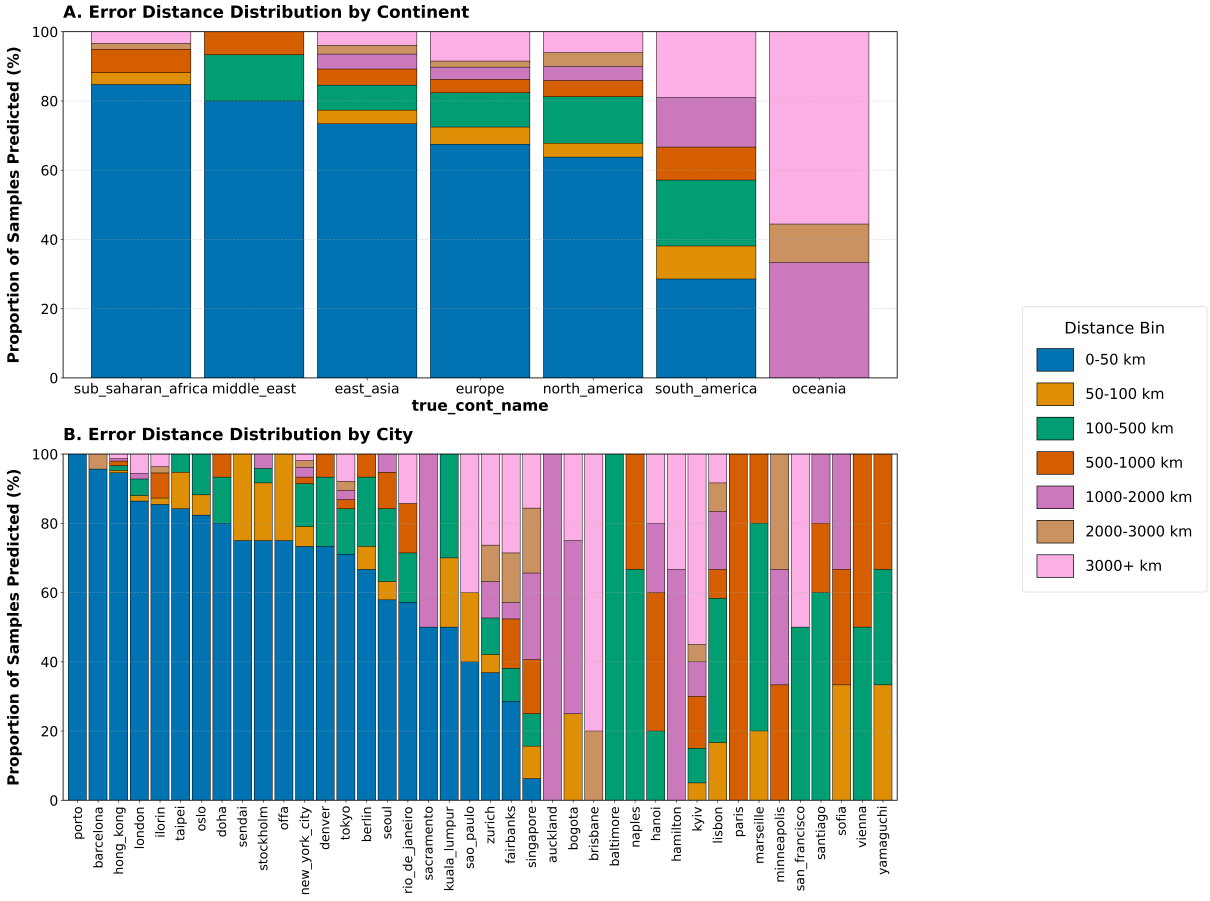

Figure 1. *Distribution of geodesic errors by continent and city for the ensemble model, showing the percentage of samples falling within various distance bins. Most predictions demonstrate high accuracy, especially for well-represented regions.*

Figure 2 visualizes the true and predicted coordinates for all test samples. The close alignment between blue (true) and red (predicted) points illustrates the high spatial accuracy achieved by the ensemble model across the globe.

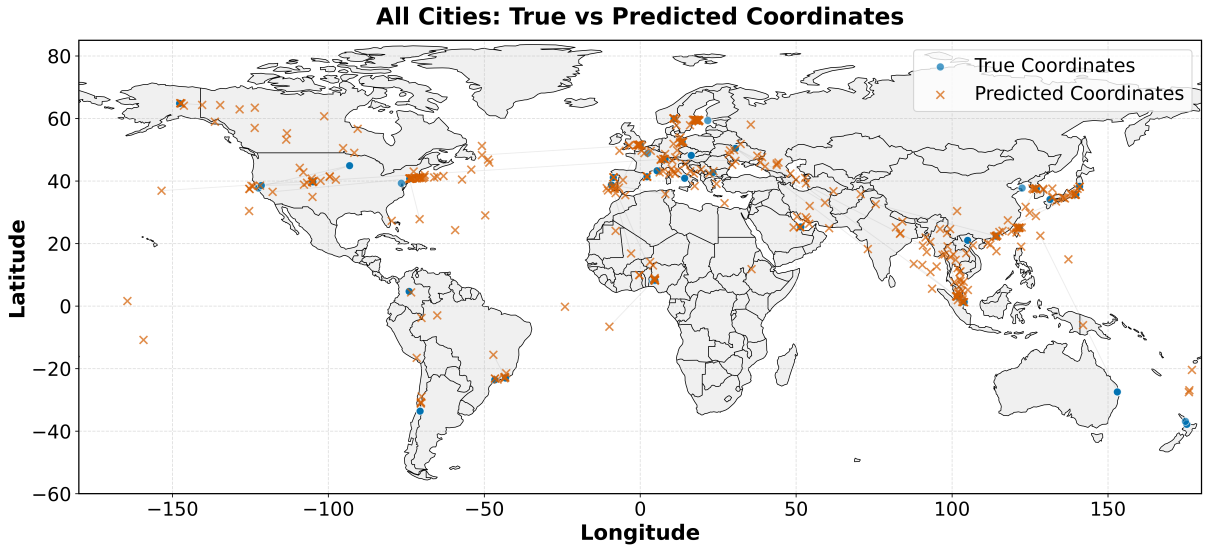

Figure 2. *World map showing the distribution of true coordinates (blue) and predicted coordinates (red) for test samples. The close alignment between true and predicted points illustrates the high spatial accuracy of the ensemble model.*

In-Radius Accuracy The in-radius accuracy metrics in Table 3 further demonstrate the ensemble model's precision. Around, 68.6% of predictions were within just 50 km of the true location, and 86.6% were within 1,000 km. These results outperform all neural network-based approaches and represent a significant increase in metagenomic geographic prediction.

Table 3. Ensemble: In-Radius Accuracy Metrics

Radius	Proportion (%)
<1 km	0.00
<5 km	4.18
<50 km	68.55
<100 km	72.85
<250 km	77.27
<500 km	81.94
<1000 km	86.61
<5000 km	96.44

1.5 Error Analysis and Hierarchical Propagation

Error group analysis for the ensemble learning model (Table 2) provides a clear understanding of how errors propagate through the prediction hierarchy (Liu et al., 2025). When both continent and city are correctly classified (Cc-Zc), the geodesic error is dramatically lower (e.g., median 12.3 km and mean 208.1 km for the ensemble model). However, errors

at the continent or city level lead to a substantial increase in geodesic error (e.g., mean error 2148.1 km for Cc-Zi, 3902.2 km for Ci-Zc, and 7365.5 km for Ci-Zi), highlighting the importance of accurate hierarchical classification for precise coordinate prediction. This underscores the need for robust models at each level of the hierarchy to minimize overall geospatial error.

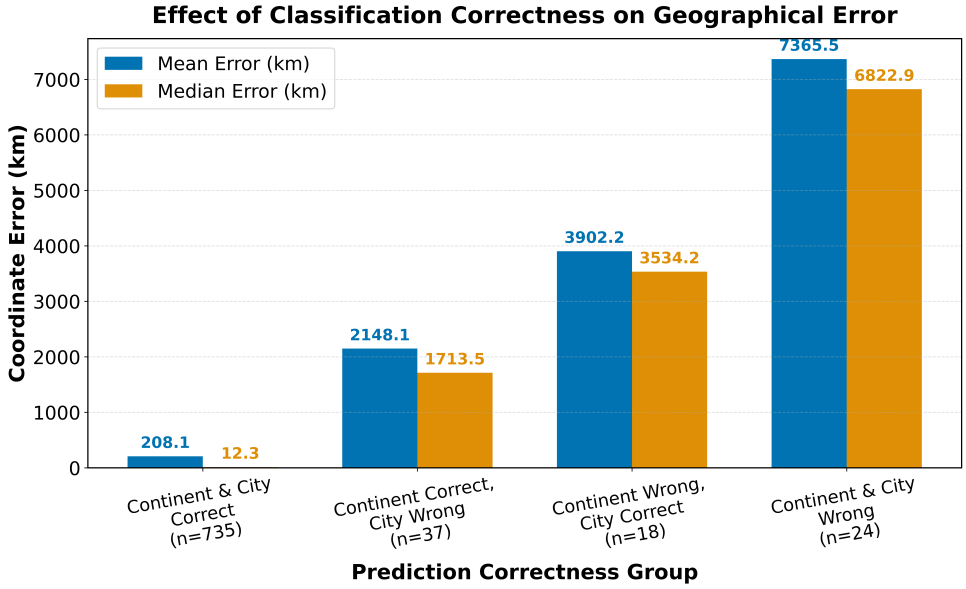

Figure 3. Classification correctness vs. geodesic error for ensemble model. The figure demonstrates the clear relationship between classification accuracy and coordinate prediction precision, with correctly classified samples showing dramatically lower geodesic errors.

1.6 Comparison with Previous State-of-the-Art (mGPS)

The mGPS (microbiome geographic population structure) tool (Zhang et al., 2024) represents the previous state-of-the-art for predicting the geographical origins of metagenomic samples from the MetaSUB dataset (Danko et al., 2021). Table 4 presents a comprehensive comparison between mGPS and our ensemble model across key performance metrics.

Table 4. Comparison of Ensemble Model and mGPS on MetaSUB Dataset

Metric	mGPS	Ensemble (TabPFN)	Notes	Reference
Sample Size	4,070 (40 cities)	4,070 (40 cities)	After QC, matched setup	–
City Prediction Accuracy	92%	93%	Test set	Supplementary Table 29
Sensitivity	78%	86.6% (Continent), 81.1% (City)	Macro-average (see Supplementary)	See text
Specificity	99%	91.7% (Continent), 85.4% (City)	Macro-average (see Supplementary)	See text
In-Radius Accuracy				
<250 km	62%	**77.27%**	Proportion of predictions within 250 km	Table 3
<500 km	74%	**81.94%**	Proportion of predictions within 500 km	Table 3
<1,000 km	84%	**86.61%**	Proportion of predictions within 1,000 km	Table 3
Median Error (km)	137	**13.72**	Median geodesic error (km)	Table 2
AUC (Continent/City)	0.99–0.996	0.928 / 0.905	OVA/OVO macro-average ROC AUC	See text
AUPR (Continent/City)	0.97 / 0.87	0.952 / 0.926	Macro-average precision-recall	See text

Notes: mGPS and Ensemble models were evaluated on the same MetaSUB dataset after quality control. City prediction accuracy, sensitivity, and specificity are reported as macro-averages on the test set. In-radius accuracy indicates the proportion of predictions within the specified geodesic distance from the true location. Median error is the median geodesic distance between predicted and true coordinates. AUC and AUPR are reported as macro-averages for continent and city classification tasks. Bold values indicate superior performance.

The ensemble model achieved a city-level accuracy of 93%, slightly surpassing mGPS (92%). More notably, it reduced the median coordinate error from 137 km (mGPS) to 13.72 km—a tenfold reduction—and increased the proportion of predictions within 250 km from 62% to 77.27%. The mean coordinate error was 589.02 km, and the 95th percentile error was 3577.48 km. While mGPS demonstrated slightly higher AUC values for classification tasks (0.99–0.996 vs. 0.928/0.905 for continent/city), our ensemble achieves comparable or superior AUPR scores (0.952/0.926 vs. 0.97/0.87 for continent/city), indicating strong performance even for imbalanced classes. Overall, our ensemble approach represents a significant advancement in the state of metagenomic geographic prediction, particularly in terms of coordinate precision and in-radius accuracy.

2. Discussion

2.1 Separate Neural Network Approach

The separate neural network models were evaluated in a hierarchical fashion: continent classification, city classification, and coordinate regression. On the test set, the continent classifier achieved an accuracy of 84.9%, with a macro F1-score of 0.78 and a weighted F1-score of 0.85 (Supplementary Table 22). At the city level, accuracy dropped to 70.1% (macro F1-score: 0.55; weighted F1-score: 0.71; Supplementary Table 26). This decrease is expected, as city-level classification involves 40 classes compared to just 7 at the continent level, making the task inherently more challenging due to increased class imbalance and finer granularity (He and Garcia, 2009). For coordinate regression, the model yielded a median geodesic error of 4,237 km and a mean error of 4,962 km, with only 1.8% of predictions within 1,000 km and 55.7% within 5,000 km of the true location (Supplementary Table 33). Regression tasks are generally more difficult than classification, especially in high-dimensional settings and with limited data (Caruana et al., 2008). These results highlight the limitations of separate neural networks for fine-grained localization in metagenomic data.

2.2 Combined Neural Network Approach

The combined hierarchical neural network jointly predicts continent, city, and coordinates using a multi-task architecture with weighted loss. This approach improved performance when compared to the separate neural networks. Continent accuracy was 82.7% (macro F1-score: 0.75; weighted F1-score: 0.83) (Supplementary Table 23), and city accuracy reached 74.9% (macro F1-score: 0.45; weighted F1-score: 0.72). This represents a 6.9% relative increase in city accuracy and a 1.4% increase in weighted F1-score over the separate neural network (Supplementary Table 27). The median geodesic error decreased from 4,237 km to 519 km (an 87.7% reduction), and the mean error dropped from 4,962 km to 1,631 km. In-radius accuracy also improved substantially, with 66.3% of predictions within 1,000 km and 89.3% within 5,000 km (Supplementary Table 33). These improvements demonstrate the benefit of joint optimization and hierarchical feature sharing, which allow information to flow between prediction tasks and mitigate error propagation. The hierarchical loss function, which jointly optimizes continent, city, and coordinate predictions, outperforms separate loss functions for each layer because it enables the model to learn shared representations and dependencies across tasks. In a hierarchical structure, errors at higher levels (e.g., continent) can propagate and negatively impact downstream predictions (e.g., city and coordinates). By optimizing a combined, weighted loss, the model is encouraged to balance performance across all levels, rather than overfitting to a single task. This joint training allows the network to leverage contextual cues and cor-

relations between tasks—such as how certain cities are only possible within specific continents—leading to more consistent and accurate predictions throughout the hierarchy. Additionally, shared feature learning reduces redundancy and improves generalization, especially in cases with limited data for fine-grained tasks. In contrast, training separate models for each layer ignores these interdependencies (Ruder, 2017).

2.3 GrowNet Model

GrowNet, a neural boosting architecture, achieved the best continent and city classification among neural models: 86.4% continent accuracy (macro F1-score: 0.77; weighted F1-score: 0.86) (Supplementary Table 24) and 75.1% city accuracy (macro F1-score: 0.60; weighted F1-score: 0.76) (Supplementary Table 28). Compared to the combined neural network, GrowNet improved city macro F1-score by 33% and weighted F1-score by 5.6%. However, at coordinate level, GrowNet achieved a median geodesic error of 823 km and mean error of 1,885 km. While GrowNet outperformed other neural models in classification, it did not match the coordinate precision of the combined neural network or ensemble models. This is due to the limited sample size, which can hinder the ability of boosting-based neural architectures to generalize in regression tasks (Zantvoort et al., 2024).

2.4 Ensemble Learning

Neural networks are known to struggle with tabular data, often failing to outperform tree-based models due to their inability to efficiently partition feature space and capture simple interactions (Grinsztajn et al., 2022a)(Grinsztajn et al., 2022b). In contrast, gradient boosting models such as XGBoost (Chen and Guestrin, 2016), LightGBM (Ke et al., 2017), and CatBoost (Prokhorenkova et al., 2018) are widely recognized as state-of-the-art for tabular data (Grinsztajn et al., 2022b). Our ensemble model integrates these algorithms with neural models and TabPFN (Hütter et al., 2022), using a hierarchical stacking approach with threshold filtering at each level.

The ensemble achieved 95% continent accuracy (macro F1-score: 0.89; weighted F1-score: 0.95) (Supplementary Table 25) and 93% city accuracy (macro F1-score: 0.80; weighted F1-score: 0.92) (Supplementary Table 29), outperforming all neural network and GrowNet models by a wide margin. Compared to GrowNet, the ensemble improved city accuracy by 17.9% and macro F1-score by 31.7%. The median coordinate error dropped to 13.72 km (from 823 km for GrowNet and 519 km for the combined neural network), and the mean error was reduced to 589.02 km. In-radius accuracy was also exceptional: 68.6% of predictions were within 50 km, 77.3% within 250 km, and 86.6% within 1,000 km. These results highlight the power of ensemble learning and the importance of leveraging diverse model types for robust, high-precision geographic prediction (Dietterich, 2000)(Opitz and

Maclin, 1999)(Mahdavi-Shahri et al., 2016).

For simple tabular datasets, gradient boosting methods like XGBoost, LightGBM, and CatBoost consistently outperform deep neural networks (Grinsztajn et al., 2022b)(Erickson et al., 2025). In our experiments, these gradient boosting models performed best at the continent and city classification stages, surpassing transformer-based models like TabPFN and neural network models such as the Separate Neural Network and GrowNet. However, as the complexity increases at the coordinate regression level, the transformer-based TabPFN model provided the most accurate predictions. Our ensemble employs threshold filtering and best-model selection at each hierarchical level (continent, city, and coordinates), ensuring that only the most reliable predictions are passed to subsequent layers.

It is important to note that these results were obtained without any hyperparameter tuning; with further optimization, we expect performance to improve.

2.5 Limitations and Future Work

Despite the substantial improvements in predictive performance, it has some notable shortcomings for our ensemble models. First among them is the very high computational demand, particularly in terms of GPU resources and runtime required to train and test the models on large datasets. This can limit scalability and accessibility for users without access to high-performance computing resources.

Going forward, research must focus on stronger and more informative feature selection. Incorporating more biological information—e.g., explicit modeling of interactions between microbial species—could lead to deeper insight into the underlying ecological processes giving rise to geographic signatures. Autoencoder-based approaches can also be utilized to extract denser, more compressed feature representations from high-dimensional data. Further improvements could be achieved by expanding the diversity of models in the ensemble, performing systematic hyperparameter optimization, and including the use of domain knowledge to guide feature engineering. Ultimately, these directions aim to enhance both the interpretability and predictive power of geographic models for metagenomic data.

2.6 Acknowledgements

I would like to thank my supervisor, Eran Elhaik, for his guidance and support throughout this project. I am also grateful to Bijan Mousavi and Sreejith for their valuable input and assistance during the course of this work.

References

Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2623–2631.

Aydin, C. C., Demir, C., and Yilmaz, E. (2016). Capability of artificial neural network for forward conversion of geodetic coordinates (phi, lambda, h) to cartesian (x,y,z) coordinates. *Environmental Earth Sciences*, 75(7):1–10.

Bergman, A. (2025). Optimizing the microbial global population structure (mgps). Unpublished manuscript, cited with permission from the author.

Caruana, R., Karampatziakis, N., and Yessenalina, A. (2008). An empirical evaluation of supervised learning in high dimensions. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, page 96–103, New York, NY, USA. Association for Computing Machinery.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794.

Danko, D., Bezdan, D., Afshin, E. E., Ahsanuddin, S., Bhattacharya, C., Butler, D. J., Chng, K. R., Donnellan, D., Hecht, J., Jackson, K., Kuchin, K., Karasikov, M., Lyons, A., Mak, L., Meleshko, D., Mustafa, H., Mutai, B., Neches, R. Y., Ng, A., Nikolayeva, O., Nikolayeva, T., Png, E., Ryon, K. A., Sanchez, J. L., Shaaban, H., Sierra, M. A., Thomas, D., Young, B., Abudayyeh, O. O., Alicea, J., Bhattacharyya, M., Blekhman, R., Castro-Nallar, E., Cañas, A. M., Chatziefthimiou, A. D., Crawford, R. W., De Filippis, F., Deng, Y., Desnues, C., Dias-Neto, E., Dybwad, M., and Elhaik, E. (2021). A global metagenomic map of urban microbiomes and antimicrobial resistance. *Cell*, 184(13):3376–3393.e17.

Dietterich, T. G. (2000). Ensemble methods in machine learning. *Multiple Classifier Systems*, 1857:1–15.

Erickson, N., Purucker, L., Tschalzev, A., Holzmüller, D., Desai, P. M., Salinas, D., and Hutter, F. (2025). Tabarena: A living benchmark for machine learning on tabular data.

Feng, J., Wang, Y., Wang, Y., Wang, Y., and Liu, Y. (2021). Grownet: Refuel boosting with concatenation and forward propagation. In *Advances in Neural Information Processing Systems*, volume 34, pages 22237–22249.

Grinsztajn, L., Oyallon, E., and Varoquaux, G. (2022a). Why do tree-based models still outperform deep learning on tabular data?

Grinsztajn, L., Oyallon, E., and Varoquaux, G. (2022b). Why do tree-based models still outperform deep learning on tabular data?

Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning*, 46(1):389–422.

He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284.

Hütter, F., Zimmer, L., Probst, P., Hees, J., Krämer, N., and Hutter, F. (2022). Tabpfn: A transformer that solves small tabular classification problems in a second. *arXiv preprint arXiv:2207.01848*.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30, pages 3146–3154.

Kosmopoulos, A., Partalas, I., Gaussier, E., Paliouras, G., and Androutsopoulos, I. (2014). Evaluation measures for hierarchical classification: a unified view and novel approaches. *Data Mining and Knowledge Discovery*, 29(3):820–865.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

Liu, H., Li, P., Hu, X., Bai, S., and Lin, Y. (2025). Multi-granularity decision information integration network for hierarchical classification via local and global constraints. *Applied Intelligence*, 55.

Mahdavi-Shahri, A., Houshmand, M., Yaghoobi, M., and Jalali, M. (2016). Applying an ensemble learning method for improving multi-label classification performance. In *2016 2nd International Conference of Signal Processing and Intelligent Systems (ICSPIS)*, page 1–6. IEEE.

Opitz, D. and Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169–198.

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., and Gulin, A. (2018). Catboost: Unbiased boosting with categorical features. *Advances in Neural Information Processing Systems*, 31:6638–6648.

Robinson, J. M., Pasternak, Z., Mason, C. E., and Elhaik, E. (2021). Forensic applications of microbiomics: A review. *Frontiers in Microbiology*, Volume 11 - 2020.

Ruder, S. (2017). An overview of multi-task learning in deep neural networks.

Snyder, J. P. (1987). *Map Projections—A Working Manual*. U.S. Geological Survey Professional Paper 1395. U.S. Government Printing Office, Washington, DC.

Tang, L. (2024). Comparison the performances for distributed machine learning: Evidence from xgboost and dnn. *Applied and Computational Engineering*, 103:209–215.

Zantvoort, K., Nacke, B., Görlich, D., Hornstein, S., Jacobi, C., and Funk, B. (2024). Estimation of minimal data sets sizes for machine learning predictions in digital mental health interventions. *npj Digital Medicine*, 7(1):361.

Zhang, Y., McCarthy, L., Ruff, E., and Elhaik, E. (2024). Microbiome geographic population structure (mgps) detects fine-scale geography. *Genome Biology and Evolution*, 16(11):evae209.

3. Supplementary Materials

3.1 Separate Neural Network Parameters

Table 5. Default parameters for separate neural network models

Parameter	Continent Model	City Model	Coordinate Model
Hidden dimensions	[128, 64]	[256, 128, 64]	[256, 128, 64]
Batch normalization	True	True	True
Initial dropout	0.3	0.3	0.2
Final dropout	0.7	0.7	0.5
Learning rate	1e-3	1e-3	1e-4
Weight decay	1e-5	1e-5	1e-5
Batch size	128	128	64
Epochs	400	400	600
Early stopping steps	20	20	30
Gradient clip	1.0	1.0	1.0

Table 6. Hyperparameter search space for neural network tuning

Hyperparameter	Search Space
Hidden dimensions	[64], [128], [128, 64], [256, 128, 64], [256, 128], [512, 256, 128, 64]
Initial dropout	0.1 to 0.3
Final dropout	0.5 to 0.8
Learning rate	1e-4 to 1e-2 (log uniform)
Batch size	64, 128, 256
Weight decay	1e-6 to 1e-3 (log uniform)
Gradient clip	0.5 to 2.0

3.2 Combined Neural Network Parameters

Table 7. Default parameters for combined neural network model

Parameter	**Value**
Architecture parameters	
Continent branch hidden dimensions	[128, 64]
City branch hidden dimensions	[256, 128, 64]
Coordinate branch hidden dimensions	[256, 128, 64]
Continent branch dropout (initial, final)	(0.3, 0.7)
City branch dropout (initial, final)	(0.3, 0.7)
Coordinate branch dropout (initial, final)	(0.2, 0.5)
Batch normalization	True
Training parameters	
Learning rate	1e-3
Weight decay	1e-5
Batch size	128
Epochs	600
Early stopping steps	50
Continent loss weight	1.0
City loss weight	0.5
Coordinate loss weight	0.2

Table 8. Hyperparameter search space for combined neural network tuning

Hyperparameter	**Search Space**
Continent branch hidden dimensions	[128, 64] or [256, 128, 64]
City branch hidden dimensions	[128, 64] or [256, 128, 64]
Coordinate branch hidden dimensions	[128, 64] or [256, 128, 64]
Continent dropout initial	0.2 to 0.5
Continent dropout final	0.6 to 0.8
City dropout initial	0.2 to 0.5
City dropout final	0.6 to 0.8
Coordinate dropout initial	0.1 to 0.3
Coordinate dropout final	0.4 to 0.6
Learning rate	1e-4 to 1e-2 (log uniform)
Weight decay	1e-6 to 1e-3 (log uniform)
Batch normalization	True or False
Batch size	64, 128, 256
Continent loss weight	1.0 to 2.0
City loss weight	0.5 to continent_weight
Coordinate loss weight	0.05 to city_weight

3.3 GrowNet Parameters

Table 9. Default parameters for hierarchical GrowNet model

Parameter	**Value**
Architecture parameters	
Hidden size	256
Input feature dimension	200
Coordinate dimension	3
Dropout rates (2 layers)	0.2, 0.4
Boosting parameters	
Number of weak learners	30
Boost rate	0.4
Epochs per stage	20
Corrective epochs	5
Training parameters	
Learning rate	1e-3
Weight decay	1e-4
Batch size	128
Early stopping steps	5
Gradient clip	1.0
Loss weights	
Continent loss weight	2.0
City loss weight	1.0
Coordinate loss weight	0.5

Table 10. Hyperparameter search space for GrowNet tuning

Hyperparameter	**Search Space**
Hidden size	128, 256, 512
Number of weak learners	10 to 30
Boost rate	0.1 to 0.8
Learning rate	1e-4 to 1e-2 (log uniform)
Batch size	64, 128, 256
Weight decay	1e-6 to 1e-3 (log uniform)
Epochs per stage	5 to 10
Gradient clip	0.5 to 2.0
Hierarchical loss weights	
Continent loss weight	1.0 to 2.0
City loss weight	0.5 to (continent_weight - 0.05)
Coordinate loss weight	0.05 to (city_weight - 0.05)

3.4 Ensemble Meta-Model Parameters

3.4.1 XGBoost Parameters

Table 11. Default parameters for XGBoost models

Parameter	**Classification**	**Regression**
Objective	multi:softprob	reg:squarederror
Eval metric	mlogloss	rmse
Learning rate	0.1	0.1
Max depth	6	6
Min child weight	1	1
Gamma	0	0
Subsample	0.8	0.8
Colsample bytree	0.8	0.8
Lambda	1.0	1.0
Alpha	0.0	0.0
n_estimators	300	300

Table 12. Hyperparameter search space for XGBoost tuning

Hyperparameter	Search Space
Learning rate	1×10^{-3} to 0.3 (log uniform)
Max depth	3 to 12
Min child weight	1 to 10
Gamma	0 to 5
Subsample	0.5 to 1.0
Colsample bytree	0.5 to 1.0
Lambda	1×10^{-3} to 10 (log uniform)
Alpha	1×10^{-3} to 10 (log uniform)
n_estimators	100 to 400

3.4.2 LightGBM Parameters

Table 13. Default parameters for LightGBM models

Parameter	Classification	Regression
Objective	multiclass	regression
Metric	multi_logloss	rmse
Learning rate	0.1	0.1
Max depth	6	6
Num leaves	31	–
Min child samples	20	20
Subsample	0.8	0.8
Colsample bytree	0.8	0.8
Reg alpha	0.1	0.0
Reg lambda	1.0	1.0
n_estimators	300	300

Table 14. Hyperparameter search space for LightGBM tuning

Hyperparameter	Search Space
Learning rate	1×10^{-3} to 0.3 (log uniform)
Max depth	3 to 12
Num leaves	15 to 256 (classification only)
Min child samples	5 to 100
Subsample	0.5 to 1.0
Colsample bytree	0.5 to 1.0
Reg lambda	1×10^{-3} to 10 (log uniform)
Reg alpha	1×10^{-3} to 10 (log uniform)
n_estimators	100 to 400

3.4.3 CatBoost Parameters

Table 15. Default parameters for CatBoost models

Parameter	Classification	Regression
Loss function	MultiClass	RMSE
Eval metric	–	RMSE
Iterations	300	300
Learning rate	0.1	0.1
Depth	6	6
L2 leaf reg	3.0	3
Random strength	–	1
Bagging temperature	–	1
Border count	–	254
Random seed	42	42
Verbose	False	False

Table 16. Hyperparameter search space for CatBoost tuning

Hyperparameter	Search Space
Iterations	100 to 400 (classification), 100 to 500 (regression)
Learning rate	1×10^{-3} to 0.3 (log uniform)
Depth	3 to 10
L2 leaf reg	1 to 10
Random strength	1×10^{-9} to 10 (log uniform, regression only)
Bagging temperature	0 to 10 (regression only)
Border count	1 to 255 (regression only)

3.4.4 GrowNet Parameters

Table 17. Default parameters for GrowNet models (ensemble context)

Parameter	**Classification**	**Regression**
Hidden size	256	256
Num weak learners	10	10
Boost rate	0.4	0.4
Learning rate	1e-3	1e-3
Weight decay	1e-5	1e-5
Batch size	128	128
Epochs per stage	30	30
Early stopping steps	7	7
Gradient clip	1.0	1.0
n_outputs	–	3

Table 18. Hyperparameter search space for GrowNet tuning (ensemble context)

Hyperparameter	**Search Space**
Hidden size	128, 256, 512
Num weak learners	10 to 30
Boost rate	0.1 to 0.8
Learning rate	1×10^{-4} to 1×10^{-2} (log uniform)
Batch size	64, 128, 256
Weight decay	1×10^{-6} to 1×10^{-3} (log uniform)
Epochs per stage	5 to 10
Gradient clip	0.5 to 2.0

3.4.5 Neural Network (MLP) Parameters

Table 19. Default parameters for neural network (MLP) models (ensemble context)

Parameter	Classification	Regression
Input dimension	200	200
Hidden dimensions	[128, 64]	[128, 64]
Output dimension	7	3
Batch normalization	True	True
Initial dropout	0.3	0.2
Final dropout	0.8	0.5
Learning rate	1e-3	1e-3
Weight decay	1e-5	1e-5
Batch size	128	128
Epochs	400	400
Early stopping steps	20	50
Gradient clip	1.0	1.0

Table 20. Hyperparameter search space for neural network (MLP) tuning (ensemble context)

Hyperparameter	Search Space
Hidden dimensions	[64], [128], [128, 64], [256, 128, 64], [256, 128], [512, 256, 128, 64]
Initial dropout	0.1 to 0.3
Final dropout	0.5 to 0.8
Learning rate	1×10^{-4} to 1×10^{-2} (log uniform)
Batch size	64, 128, 256
Weight decay	1×10^{-6} to 1×10^{-3} (log uniform)
Gradient clip	0.5 to 2.0

3.4.6 TabPFN Parameters

Table 21. TabPFN model configuration

Parameter	Value
Model	Pre-trained TabPFN
Hyperparameter tuning	Max time

3.5 Continent Classification: Separate Neural Network

Table 22. Continent Classification Report (Separate Neural Network)

Continent	Precision	Recall	F1-score	Support
east_asia	0.93	0.89	0.91	278
europe	0.86	0.82	0.84	283
middle_east	0.93	0.93	0.93	15
north_america	0.74	0.85	0.79	149
oceania	0.31	0.44	0.36	9
south_america	0.75	0.71	0.73	21
sub_saharan_africa	0.88	0.88	0.88	59
Accuracy		0.85 (814 samples)		
Macro avg	0.77	0.79	0.78	814
Weighted avg	0.86	0.85	0.85	814

3.6 Contienent Classification: Combined Neural Network

Table 23. Continent Classification Report (Combined Neural Network)

Continent	Precision	Recall	F1-score	Support
east_asia	0.90	0.90	0.90	278
europe	0.89	0.74	0.81	283
middle_east	0.70	0.93	0.80	15
north_america	0.72	0.85	0.78	149
oceania	0.33	0.44	0.38	9
south_america	0.65	0.81	0.72	21
sub_saharan_africa	0.80	0.90	0.85	59
Accuracy		0.83 (814 samples)		
Macro avg	0.71	0.80	0.75	814
Weighted avg	0.84	0.83	0.83	814

3.7 Continent Classification: Hierarchical GrowNet

Table 24. Continent Classification Report (GrowNet)

Continent	Precision	Recall	F1-score	Support
east_asia	0.94	0.94	0.94	278
europe	0.87	0.81	0.84	283
middle_east	0.70	0.93	0.80	15
north_america	0.75	0.87	0.80	149
oceania	0.29	0.22	0.25	9
south_america	1.00	0.81	0.89	21
sub_saharan_africa	0.89	0.85	0.87	59
Accuracy		0.86 (814 samples)		
Macro avg	0.78	0.78	0.77	814
Weighted avg	0.87	0.86	0.86	814

3.8 Continent Classification: Ensemble Learning

Table 25. Continent Classification Report (Ensemble Learning)

Continent	Precision	Recall	F1-score	Support
east_asia	0.95	0.97	0.96	278
europe	0.95	0.94	0.95	283
middle_east	0.93	0.93	0.93	15
north_america	0.93	0.97	0.95	149
oceania	0.67	0.44	0.53	9
south_america	1.00	0.86	0.92	21
sub_saharan_africa	0.98	0.95	0.97	59
Accuracy		0.95 (814 samples)		
Macro avg	0.92	0.87	0.89	814
Weighted avg	0.95	0.95	0.95	814

3.9 City Classification: Separate Neural Network

Table 26. City-level classification report for Separate Neural Network on the test set.

City	Prec.	Rec.	F1	Sup.
auckland	0.00	0.00	0.00	1
baltimore	0.33	1.00	0.50	1
barcelona	0.96	1.00	0.98	23
berlin	0.50	0.93	0.65	15
bogota	0.67	0.50	0.57	4
brisbane	0.40	0.80	0.53	5
denver	0.54	0.87	0.67	15
doha	0.93	0.93	0.93	15
europe	0.59	0.83	0.69	12
fairbanks	0.50	0.24	0.32	21
hamilton	0.25	0.33	0.29	3
hanoi	0.75	0.60	0.67	5
hong_kong	0.98	0.86	0.92	148
ilorin	0.87	0.62	0.72	55
kuala_lumpur	0.69	0.90	0.78	10
kyiv	0.42	0.50	0.45	20
lisbon	0.38	0.25	0.30	12
london	0.91	0.64	0.75	125
marseille	0.80	0.80	0.80	5
minneapolis	1.00	0.33	0.50	3
naples	0.67	0.67	0.67	3
new_york_city	0.72	0.83	0.77	105
offa	0.10	0.50	0.17	4
oslo	0.52	0.94	0.67	17
paris	0.00	0.00	0.00	1
rio_de_janeiro	0.83	0.71	0.77	7
sacramento	0.50	1.00	0.67	2
san_francisco	0.25	0.50	0.33	2
santiago	0.83	1.00	0.91	5
sao_paulo	0.40	0.40	0.40	5
sendai	0.33	1.00	0.50	4
seoul	0.77	0.89	0.83	19
singapore	0.45	0.31	0.37	32
sofia	0.50	0.67	0.57	3
stockholm	0.64	0.29	0.40	24
taipei	0.76	1.00	0.86	19
tokyo	0.67	0.53	0.59	38
vienna	0.00	0.00	0.00	4
yamaguchi	0.00	0.00	0.00	3
zurich	0.46	0.58	0.51	19
accuracy			0.70	814
macro avg	0.55	0.62	0.55	814
weighted avg	0.75	0.70	0.71	814

3.10 City Classification: Combined Neural Network

Table 27. City-level classification report for Combined Neural Network on the test set.

City	Prec.	Rec.	F1	Sup.
auckland	0.00	0.00	0.00	1
baltimore	0.00	0.00	0.00	1
barcelona	0.96	1.00	0.98	23
berlin	1.00	0.13	0.24	15
bogota	0.00	0.00	0.00	4
brisbane	0.00	0.00	0.00	5
denver	0.76	0.87	0.81	15
doha	0.70	0.93	0.80	15
europe	0.39	1.00	0.56	12
fairbanks	0.75	0.43	0.55	21
hamilton	0.00	0.00	0.00	3
hanoi	0.00	0.00	0.00	5
hong_kong	0.94	0.99	0.96	148
ilorin	0.76	0.95	0.85	55
kuala_lumpur	0.78	0.70	0.74	10
kyiv	1.00	0.05	0.10	20
lisbon	0.33	0.17	0.22	12
london	0.94	0.74	0.83	125
marseille	0.00	0.00	0.00	5
minneapolis	0.00	0.00	0.00	3
naples	0.00	0.00	0.00	3
new_york_city	0.70	0.91	0.79	105
offa	0.00	0.00	0.00	4
oslo	0.58	0.82	0.68	17
paris	1.00	1.00	1.00	1
rio_de_janeiro	0.57	0.57	0.57	7
sacramento	0.50	0.50	0.50	2
san_francisco	0.50	1.00	0.67	2
santiago	0.83	1.00	0.91	5
sao_paulo	0.43	0.60	0.50	5
sendai	1.00	0.25	0.40	4
seoul	0.85	0.89	0.87	19
singapore	0.43	0.75	0.55	32
sofia	0.00	0.00	0.00	3
stockholm	0.87	0.54	0.67	24
taipei	0.90	1.00	0.95	19
tokyo	0.63	0.71	0.67	38
vienna	0.00	0.00	0.00	4
yamaguchi	0.00	0.00	0.00	3
zurich	0.53	0.53	0.53	19
accuracy			0.75	814
macro avg	0.49	0.48	0.45	814
weighted avg	0.75	0.75	0.72	814

3.11 City Classification: Hierarchical GrowNet

Table 28. City-level classification report for Hierarchical GrowNet on the test set.

City	Prec.	Rec.	F1	Sup.
auckland	0.00	0.00	0.00	3
baltimore	0.00	0.00	0.00	0
barcelona	1.00	0.95	0.97	19
berlin	0.64	0.93	0.76	15
bogota	1.00	0.50	0.67	2
brisbane	0.33	0.50	0.40	4
denver	0.62	0.62	0.62	13
doha	0.74	0.93	0.82	15
europe	0.76	0.72	0.74	18
fairbanks	0.32	0.39	0.35	18
hamilton	0.00	0.00	0.00	2
hanoi	0.38	1.00	0.55	3
hong_kong	0.97	0.86	0.91	179
ilorin	0.91	0.74	0.81	53
kuala_lumpur	0.85	0.92	0.88	12
kyiv	0.19	0.46	0.27	13
lisbon	0.26	0.31	0.29	16
london	0.88	0.76	0.82	123
marseille	0.71	1.00	0.83	5
minneapolis	0.25	1.00	0.40	1
naples	1.00	0.20	0.33	5
new_york_city	0.75	0.74	0.75	105
offa	0.00	0.00	0.00	6
oslo	0.77	0.85	0.81	20
paris	0.33	0.50	0.40	2
rio_de_janeiro	1.00	0.67	0.80	6
sacramento	1.00	0.67	0.80	6
san_francisco	0.56	0.83	0.67	6
santiago	0.80	0.80	0.80	5
sao_paulo	1.00	0.75	0.86	8
sendai	0.67	1.00	0.80	6
seoul	0.71	1.00	0.83	15
singapore	0.59	0.42	0.49	24
sofia	0.33	0.50	0.40	2
stockholm	0.88	0.85	0.87	27
taipei	0.87	1.00	0.93	13
tokyo	0.73	0.70	0.71	23
vienna	0.50	1.00	0.67	1
yamaguchi	0.33	0.33	0.33	3
zurich	0.71	0.59	0.65	17
accuracy			0.75	814
macro avg	0.61	0.65	0.60	814
weighted avg	0.79	0.75	0.76	814

3.12 City Classification: Ensemble Learning

Table 29. City-level classification report for Ensemble Learning on the test set.

City	Prec.	Rec.	F1	Sup.
auckland	0.33	1.00	0.50	1
baltimore	0.00	0.00	0.00	1
barcelona	1.00	1.00	1.00	23
berlin	0.94	1.00	0.97	15
bogota	1.00	0.75	0.86	4
brisbane	1.00	0.60	0.75	5
denver	0.94	1.00	0.97	15
doha	1.00	0.93	0.97	15
fairbanks	0.83	0.95	0.89	21
hamilton	1.00	0.67	0.80	3
hanoi	1.00	0.80	0.89	5
hong_kong	0.99	0.99	0.99	148
ilorin	0.98	0.93	0.95	55
kuala_lumpur	0.91	1.00	0.95	10
kyiv	0.58	0.70	0.64	20
lisbon	0.92	0.92	0.92	12
london	1.00	0.97	0.98	125
marseille	0.75	0.60	0.67	5
minneapolis	0.60	1.00	0.75	3
naples	0.67	0.67	0.67	3
new_york_city	0.95	0.97	0.96	105
offa	0.67	1.00	0.80	4
oslo	1.00	0.94	0.97	17
paris	0.00	0.00	0.00	1
porto	0.92	1.00	0.96	12
rio_de_janeiro	1.00	0.86	0.92	7
sacramento	1.00	1.00	1.00	2
san_francisco	0.67	1.00	0.80	2
santiago	1.00	1.00	1.00	5
sao_paulo	1.00	0.60	0.75	5
sendai	1.00	1.00	1.00	4
seoul	0.86	0.95	0.90	19
singapore	0.73	0.84	0.78	32
sofia	1.00	0.67	0.80	3
stockholm	0.96	1.00	0.98	24
taipei	0.90	1.00	0.95	19
tokyo	0.85	0.87	0.86	38
vienna	0.60	0.75	0.67	4
yamaguchi	0.00	0.00	0.00	3
zurich	0.91	0.53	0.67	19
accuracy			0.93	814
macro avg	0.81	0.81	0.80	814
weighted avg	0.93	0.93	0.92	814

3.13 Coordinate Regression: Separate Neural Network

Table 30. Error Group Analysis (Separate Neural Network)

Group	Count	Mean Error (km)	Median Error (km)	Proportion	Weighted Error
C_correct Z_correct	565	3994	3255	0.694	2772
C_correct Z_wrong	126	5333	3703	0.155	826
C_wrong Z_correct	6	7668	8555	0.007	57
C_wrong Z_wrong	117	9098	7532	0.144	1308

Notes: C = Continent, Z = City. Groups indicate correctness of continent and city predictions.

3.14 Coordinate Regression: Combined Neural Network

Table 31. Error Group Analysis (Combined Neural Network)

Group	Count	Mean Error (km)	Median Error (km)	Proportion	Weighted Error
C_correct Z_correct	581	502	274	0.714	358
C_correct Z_wrong	92	2101	1523	0.113	237
C_wrong Z_correct	29	3434	2252	0.036	122
C_wrong Z_wrong	112	6637	5377	0.138	913

Notes: C = Continent, Z = City. Groups indicate correctness of continent and city predictions.

3.15 Coordinate Regression Metrics: Hierarchical GrowNet

Table 32. Error Group Analysis (GrowNet)

Group	Count	Mean Error (km)	Median Error (km)	Proportion	Weighted Error
C_correct Z_correct	604	904	599	0.742	671
C_correct Z_wrong	99	2215	1710	0.122	269
C_wrong Z_correct	7	4501	4324	0.009	39
C_wrong Z_wrong	104	7090	5896	0.128	906

Notes: C = Continent, Z = City. Groups indicate correctness of continent and city predictions.

3.16 In-Radius Accuracy Metrics

Table 33. In-Radius Accuracy Metrics for Separate Neural Network, Combined Neural Network, and Hierarchical GrowNet on the test set.

Radius	Separate NN (%)	Combined NN (%)	GrowNet (%)
<1 km	0.00	0.00	0.00
<5 km	0.00	0.00	0.00
<50 km	0.00	0.37	0.98
<100 km	0.00	9.46	2.70
<250 km	0.00	30.34	12.78
<500 km	0.86	49.75	30.96
<1000 km	1.84	66.34	57.37
<5000 km	55.65	89.31	89.07