Approach to Solve the Problem

The dataset provided has initially 9 features.
In the final submission, some of the features was removed and some were added and the final dataset had 10 features.

The initial approach or the first cut solution was to :

- Perform imputation if missing value were present in dataset.
- Perform EDA- univariate analysis like PDFs, Histograms, Box plots etc, followed by multivariate plots like pair plot, Correlation matrix .
- Performing EDA would give ideas for feature engineering.
- Perform feature engineering - based on the feature names of the dataset, some FE ideas like number of views per followers and trying out some interaction variables were on the mind.
- Perform train-test split to evaluate the models.
- Try out various ML models starting from simple models like Linear regression and based on the performance, move to more complex models.
- Perform hyperparameter tuning Models

Data Preprocessing / feature Engineering

- Several feature engineering ideas were tried like transformational features like log(x), x^c, exp(x), 1/(x), boxcox(x). These were tried out to different features and at the end boxcox(views) showed improvement in validation R2 score.
- Other interaction variables tried out were followers/age, views/followers, followers/age and some other slightly complex interaction variables. From those only ‘views/followers’ showed improvement in performance.
- Trying to add more new features caused a decrease in r2 score. Hence only these 2 features were introduced.
- Autoencoders were also tried out as feature engineering strategy. But the obtained features did not show improvement in the test performance and hence were not used.

Final Notebook

- Coming to Modelling, the simpler models like Logistic regression, KNN regressor, SVR all showed low R2 score.
- More complex models like RandomForest, XGBoost, LightGBM and a 5 layer neural network, all showed almost similar performance. After all the models stacking regressor was also tried out using the best performed models as base learners.
- Out of all these models, XGBoost showed slighlty better performance.
- Then hyperparmaeter tuning of XGBoost was tried out using Optuna. But the parameters from this was found to be not the best in terms of performance. Hence after some trial and error the optimum parameters were found out and the final test R2 score of 0.4747 was obtained.