Approach:

==========

I have started with classical ML models fitting entire train data after the split being done randomly(80:20).Once evaluation started the model was giving pretty bad R2 score though we have tried with different ML models like Ridge/Lasso/ElasticNet to check if the data is having any linear pattern. But we observed while performing EDA that the data is not linear. Later we tried with tree based algo's like DT/different ensemble models which improved the r2 score but not significantly.

Later I thought of going with a clustering approach to group the data based upon engagement score(0-1/1-2/2-3/3-4/4-5) but any clustering algo's were giving the proper grouping. Later I thought of going with custom clustering/segmenting the data based upon engagement score.

As part of featurization of the data, as most of the features are categorical, I used a countvectorizer(OHE) to transform the data into vector.As part of the feature engineering, I tried with followers/view ratio but didn't enhance the performance much.

I created a feature called eng_score_bracket where I grouped the part of data based upon its engagement score, like if the datapoint is having engagement score of 0.5 it should go to bracket of 1, likewise 1.5 to 2 and vice versa.

Later I trained a single model on every set of data(5 data set - df1,df2,df3,df4,df5 according to script) and created 5 models which trained on different set of data.

I created one driver dataframe copying it from actual train-data provided in the competition.The test data given in the competition to predict, iterate every datapoint in it and used the driver table to flow to specific model based upon User-id, category-id,video-id so that it should predict the nearest predicted value.

As thinking about the cold start problem based upon user-id/video-id, we have maintained one flow based upon category id.

By applying this approach I enhanced my prediction by almost 60% in the test data split from the train data set and almost 43% increase in the public test data which was way higher than the classical ML approach.