Reflection Report

Column-wise summary

CommitType

For this I had 2 options

1. Applying Label Encoder

Like applying 1 for feature,2 for bugfix, 3 for refractor and so on.. But there is a problem that it gives priority to certain high numbers in somehow ways

2 Applying OHE

I guess this is the best thing to do because it also increase the dimensions and basically it helps the model as well if i used random forest

FileExtension

I had many options

1. OHE

Applying ohe on each extension like

Advantages:-

There will be a lot many columns and maybe they will disturbing

Disadvantages:-

Good for trees model as they love columns

2. Vectorization Count

Applying vectorization method like *bag of words* or *tf-idf* with ngrams of 1,2 i will try to convert ['py' , 'html', 'css'] something like this and then apply this all

Advantages:-

Simple and manages frequency also but there will be no repeating extension as seen in EDA

Disadvantages:-

Less interpretable because extension are not always same but still can be tried if base model accuracy is not upto mark

3. Group Flags

This is what i think is proper because grouping in terms of role category will really help a lot like grouping in category of words like *has_frontend, has_backend, has_db, has_docs, num_ext* will haelp a lot over here

Advantages :- Very interpretable

Disadvantages :- need to create function as py and java both in backend

Priority

1. Group Flags
2. Vectorization
3. OHE

Made a function to extract the words I made it with the help of chatgpt

Numerical Columns

There were no missing values good to know

Mean of linesadded us max

Variance of linesadded is more

Changes I did

num of fileschanged is small so no change

lines added is not skewed like tail has high values I will do log scaled

lines deleted same problem

numcommentsadded same problem

Created extra columns like net_lines,churn,avg_added_file,avg_deleted_file,comment_ratio,

Time Columns

I extracted date day and also what i thought if there is a column of duration because most of developers submit their code before night and qa works at night i gave a try

CommitMessage

Had three options

1. Vectorization in tf-idf (ig this is best)
2. Vectorization in bagofwords (not bad ig)
3. Doing OHE with topcount of 50 words(worst thing ig)

downloaded stop_words,lowered text. Split text and then cleaned it

Used TF-IDF with ngrams (1,2) Used help of chatgpt to write this part of code

Had two options Stemming and Lemmatization

Used LabelEncoder in TargetColumn

MODEL TRAINING

Though of using LR with RandomForest but started with LGBMClassifier

And also tried LR, Random Forest, XGboost

On comparing all I got :-

```
{'lgbm': 0.9833333333333333, 'lr': 0.7633333333333333, 'rf': 0.9833333333333333, 'xgb': 0.9833333333333333}
```
