Data analysis of the site "SberAutopodpiska"

•••

analysis and prediction of targeted user actions

Mykhailo Kafka
mykhailokafka@gmail.com

Targets and goals

Exploratory data analysis

Data Acquaintance

Assessment of completeness and purity

Basic cleaning of duplicates, gaps

Estimating Distributions and Relationships

Create and train a model for:

Target action group predictions

Target metric ROC-AUC > 0.65

Pack the model into a service that accepts input

all signs of visit_*, utm_*, device_*, geo_*

and returning 0 or 1

where 1 - the user has completed the target action

Project realization

At the input files with sessions and events:

ga_sessions.csv - sessions:

- 1.8 million objects
- 18 signs
- 4 columns give numerical features
- 3 variables are useless
- 11 categorical features

ga_hits.csv - events:

- 15.7 million objects
- 11 signs
- event_action - target variable
- 2.7% of sessions with targeted action

When there are more targeted actions:

In the daytime

At the beginning of the week

On repeat visits

Not from a social network

From organic traffic

From a computer

From Moscow and the region

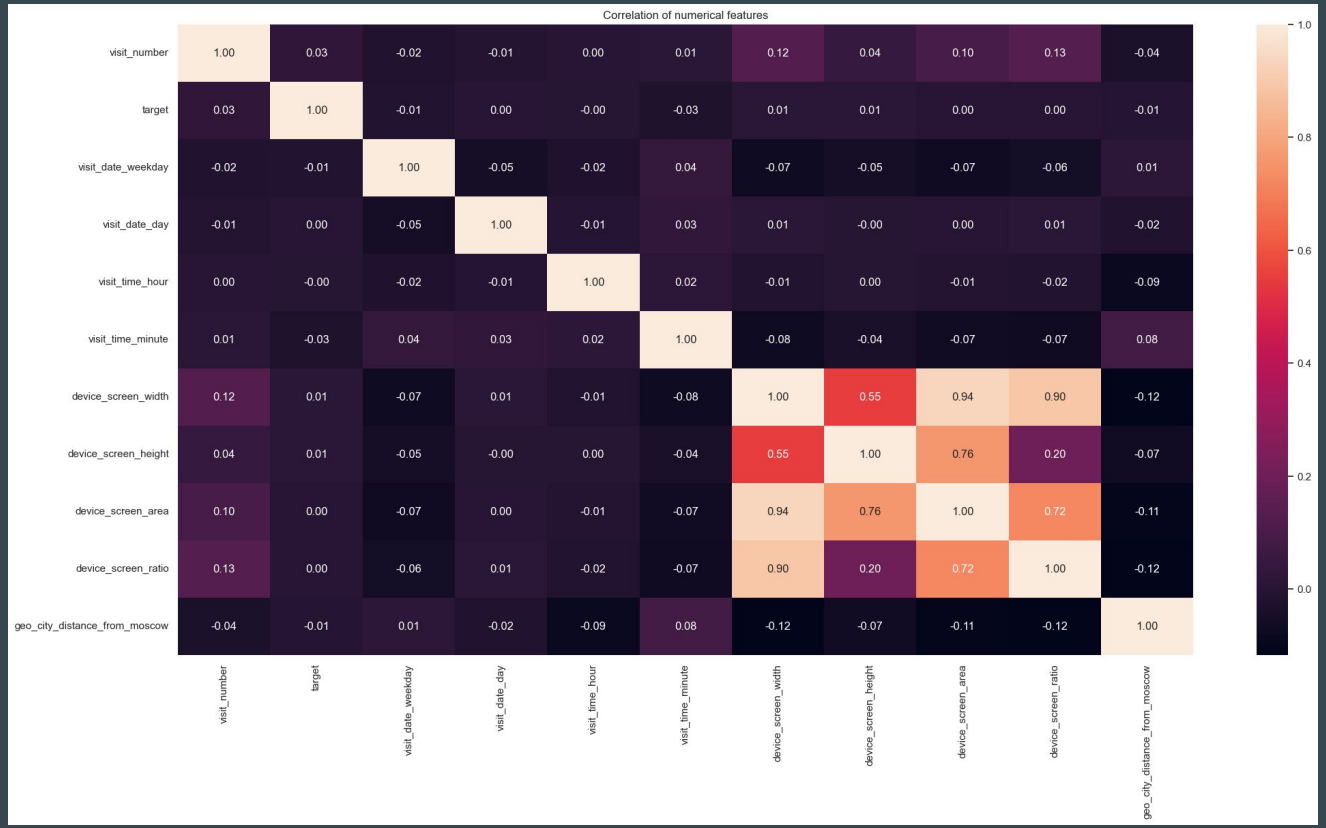
Target actions for visits:

4+	4.4%
3	3.8%
2	3.2%
1	2.4%

Additional features

From date and time	day of the week and hour
Organic traffic	
Traffic from social networks	
From screen size	width, screen area
For cities	Moscow region
Distance to Moscow	

Correlation of numerical features



Data preparation

Creating features

Numerical conversions

Categorical transformations

Feature Selection

```
('indexer', FunctionTransformer(set_index)),
('imputer', FunctionTransformer(fill_missings)),
('engineer', FunctionTransformer(create_features)),
('dropper', DropFeatures([...])),

('normalization', YeoJohnsonTransformer()),
('outlier_remover', Winsorizer()),
('scaler', StandardScaler()),

('rare_encoder', RareLabelEncoder(tol=0.05, replace_with='rare')),
('onehot_encoder', OneHotEncoder(drop_last_binary=True)),
('bool_converter', FunctionTransformer(converse_types)),

('constant_dropper', DropConstantFeatures(tol=0.99)),
('duplicated_dropper', DropDuplicateFeatures()),
('correlated_dropper', DropCorrelatedFeatures(threshold=0.8)),
```

LightGBM is the best model

- High ROC-AUC
- Fast learning
- Interpreted
- Predicts probability

Model	ROC-AUC
Histogram boosting	0.7070
LightGBM	0.7066
CatBoost	0.7062
Neural network	0.70
XGBoost	0.69
Logistic Regression	0.67
Bayes classifier	0.65
Random Forest	0.63
Support vector classifier	0.62
Decision Tree	0.52

Model optimization

4700 trees

16 leaves per tree

0.04 learning rate

GBDT boosting type

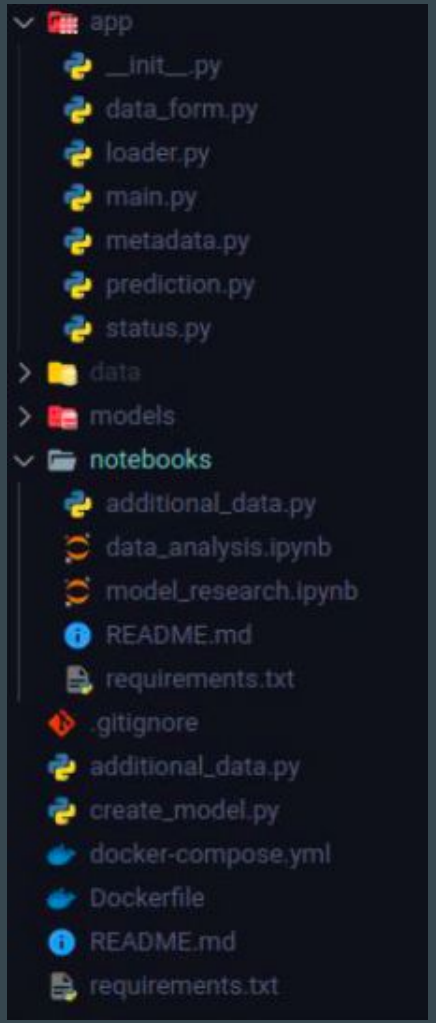
Service creation

Service in a separate module

Model creation - separately

FastAPI + Uvicorn + Pydamik

Packaging in Docker

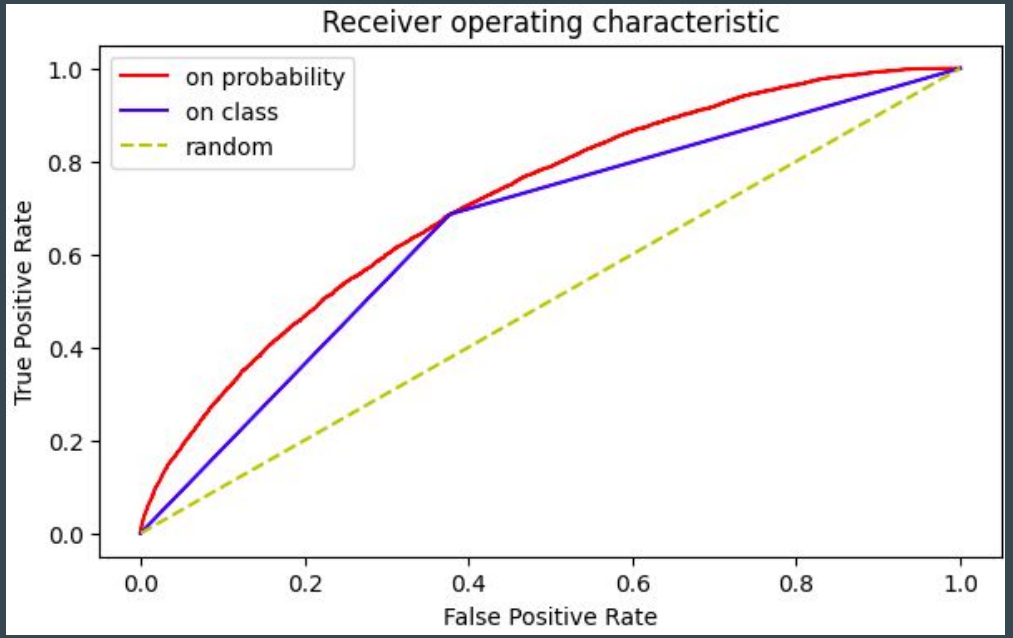


Model Metrics

ROC-AUC	0.715
ROC-AUC (classes)	0.653
ACCURACY	0.563
PRECISION	0.045
RECALL	0.748
F1	0.085
Probability Threshold	0.0239

Error Matrix:

prediction| true label|	0.0	1.0
0.0	121162	73428
1.0	1698	3712



Data processing results

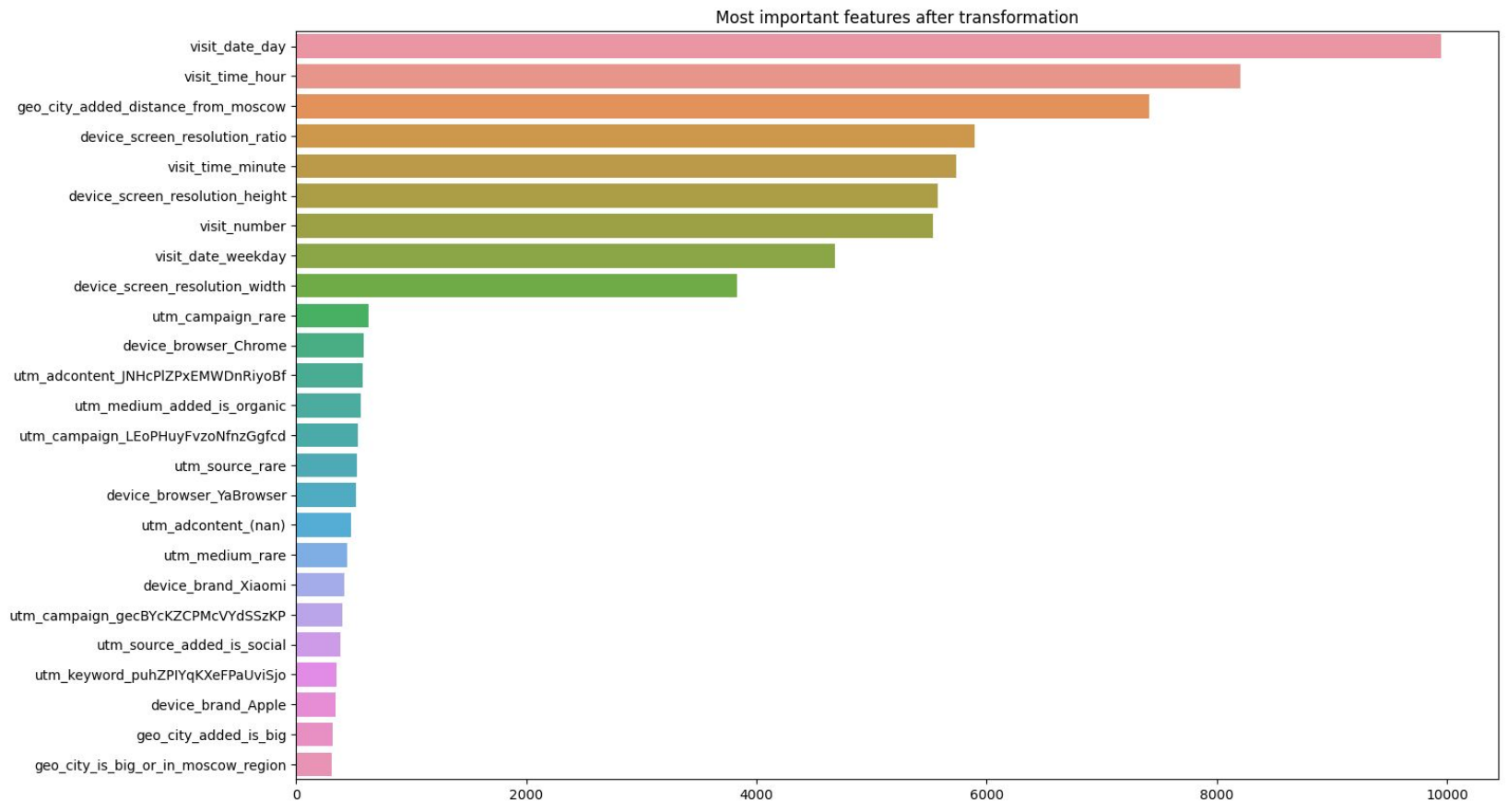
62 signs (16 removed)

9 - numerical

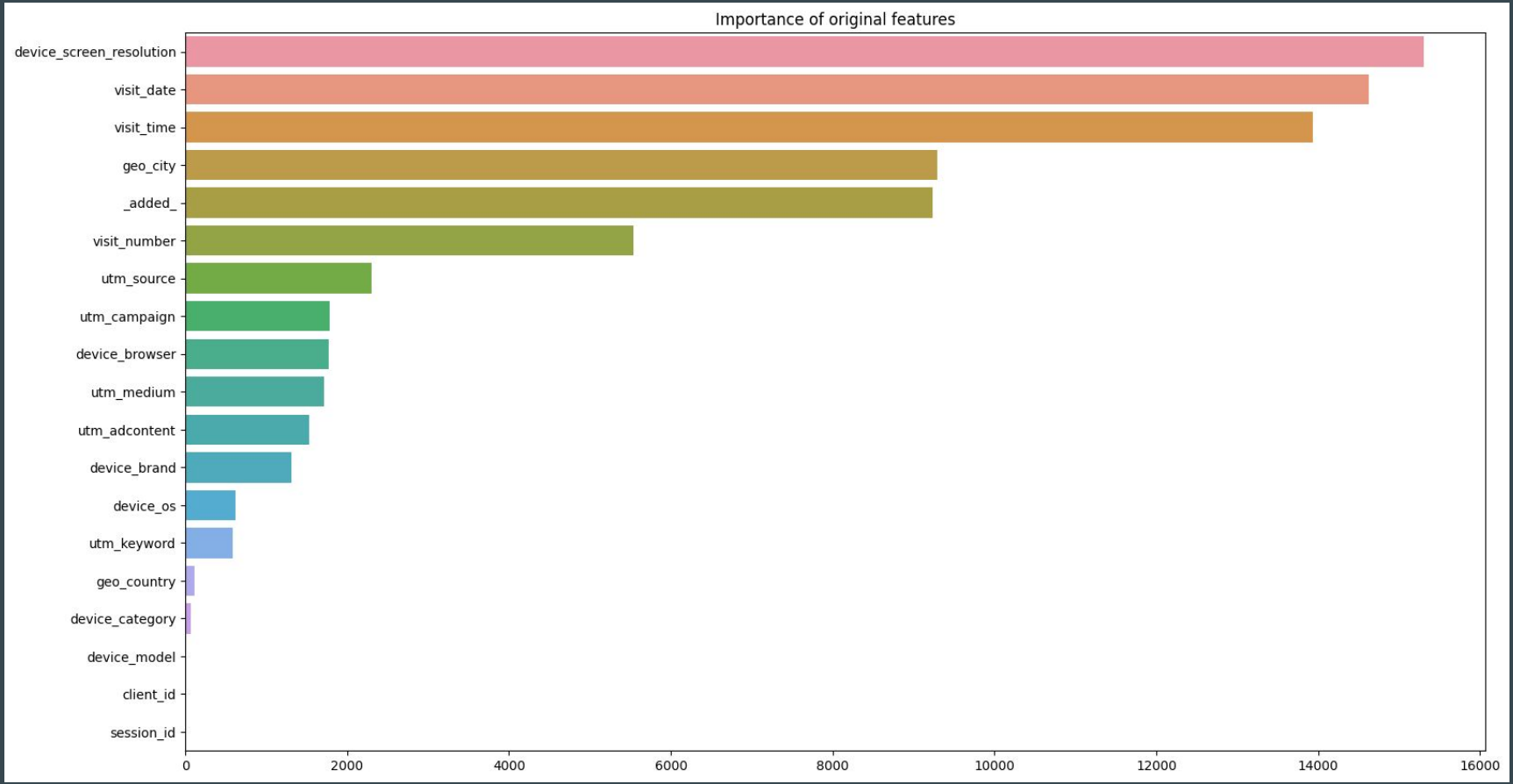
175,000 duplicates

No correlations with target variable

The most important features created



Importance of original features



Service operation

Work via Uvicorn or Docker

Prediction time 1-2 seconds

Returns status and metadata

Class or probability prediction

For one object and for many