BilDeal

[Jakob Bang and Emil Molberg / Bildeal Inc (group 35)], [15/11/2024]

DESCRIBE THE PROBLEM

SCOPE

- Objective: To estimate the value of a used car.
- The results/solution in the project can be used by anyone to determine a car’s value, which would prove instrumental in buying a used car and getting your money’s worth out of it. Per today, this problem is usually solved by going to the seller of the car and inspecting the car yourself, alternatively one can procure pictures of said car, however that wouldn’t necessarily produce accurate depictions.
- Performance can be measured using the first contact resolution rate, customer satisfaction and feedback, user engagement rates and content diversity.
- The components of the system will primarily consist of the website (UI) and the ai program itself. As the website serves as the user interface, changes should work in tandem with the ai setup we have made.
- The stakeholders of the project would be people wanting to know the price of a used car. Usually this is a core part of buying a used car, and such knowledge can prove invaluable.

METRICS

For the project to be successful, the website needs to produce accurate numbers pertaining to the anticipated price of a used car. For that to happen the metrics we provide for the ai also need to be accurate and accounted for.

Machine Learning Metrics

1. **Model Stability Metrics:**
   - Cross-Validation Score: Evaluate the model across multiple folds of the data to ensure it generalizes well.
   - Prediction Consistency: Ensure predictions vary minimally for slight changes in input data.

2. **Feature Importance**:
   - Feature importance scores from tree-based model (random forest regressor) to understand which features influence the price most. Result showed that Mileage (37%), Model year (16%) and horsepower (10%) was the most important.

System Performance Metrics

1. **Latency**:
   - Time taken for the system to return a prediction.
   - Threshold: Ideally <500 ms for real-time applications.

2. **Throughput**:
   - Number of predictions the system can handle per second.
   - Threshold: Should support expected user load (e.g., 50 predictions/sec for a small-scale deployment).

3. **Error Rate**:
   - Percentage of failed predictions (e.g., due to invalid input or system downtime).
   - Threshold: <1% error rate.

4. **Uptime**:
   - Measure of system availability (e.g., percentage of time the API or website is operational).
   - Threshold: >99.9% uptime.

DATA

Privacy and Ethical considerations:

- Privacy regulations according to GDPR and CCPA.
- Avoiding collection of personal identifiable information, such as names, contact details and addresses.
- Bias and fairness. No overrepresentation or underrepresentation of certain car brands.
- Transparency. Be open about data sampling.
- Usage ethics. Avoid applications that could manipulate car pricing to disadvantage consumers unfairly.

Data sources:

- Kaggle

Representation: Numerical features such as car age and mileage will be represented as numerical values. Categorical features will be encoded using one-hot encoding.

Data cleaning: We handle missing values by using imputation, putting in the median based on the datasets.

Ground Truth Validation:

- Explain any potential collaboration with dealerships or platforms to validate prediction outputs against real sales data.
- Incorporate data sampling techniques to prevent noisy data from corrupting ground truth labels.

Ensuring Accuracy and Consistency:

- normalize units (e.g., mileage in miles, price in Nok)
- Remove outliers using statistical methods (e.g. Z-score or IQR)
- Regularly audit the dataset for anomalies.

Features (inputs):

- Car Age (years):
- Mileage (miles)
- Brand (categorical)
- Fuel type (categorical)
- Model type (categorical)
- Transmission (categorical
- Engine (HP)
- Number of owners (numerical)
- Previous accidents (numerical)
- Clean title (Boolean)

Label (output)

- Estimated car price (NOK)

MODELING

1. **Baseline Performance**:
   - We're using a decision tree regressor to get our expected price, then afterwards we match them up with a various number of websites to check if the numbers align. As there's a certain amount of human subjectiveness when it comes to selling used cars, we can assume that it won't always be 100% accurate.

2. **Error Analysis**:
   - We fix errors manually

DEPLOYMENT

How will the model(s) be deployed? How will the predictions be used? What are your plans for monitoring and maintaining the machine learning system? If relevant, how do you plan to improve the system after deployment?

1. **Monitoring & Maintenance**:

   We primarily deploy and use the program when testing is needed locally.

2. **User Feedback Loop**:
   The only real feedback loop that we have is when we manually match up the prices given by the program and match them up with real life prices.

REFERENCES

Kaggle.com (15/11/2024), datasets