MACHINE LEARNING

Home About Show

COMPUTER

COMPUTER PRICE ESTIMATION PRESENTATION

Get Started Make an estimation

Welcome to Presentation

EDA

Title: Data Cleaning & Preparation Content:

•**Missing Value Handling:**

Numeric features were filled using the median strategy.

Categorical features were filled using the mode strategy.

•**Data Formatting:** Ensured release_year and other discrete variables were formatted correctly for analysis.

•**Feature Selection:** Selected 16 key numeric features (e.g., cpu_base_ghz, ram_gb, warranty_months) for the initial regression model.

EDA

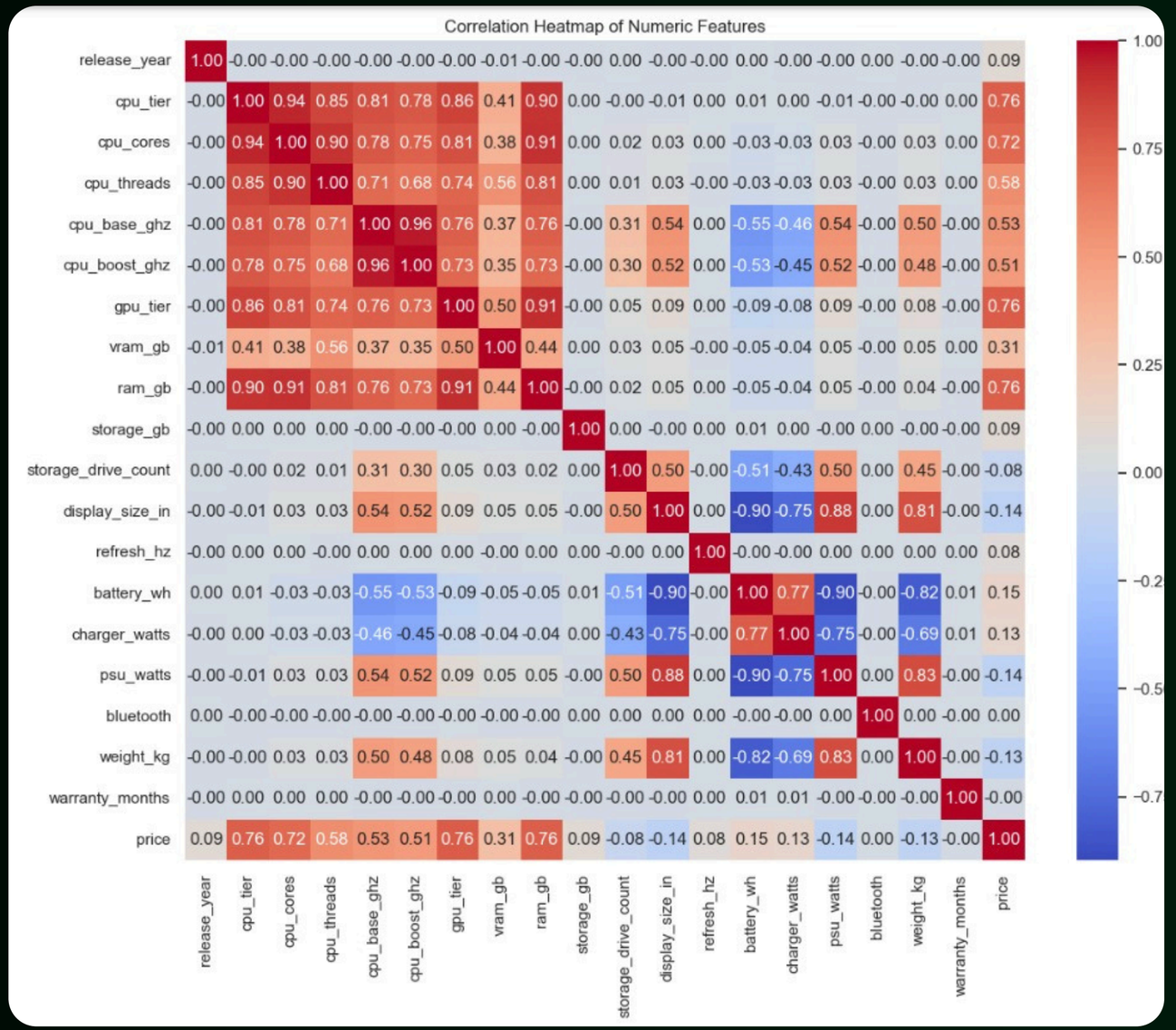
Title: Visualizing Data Distribution & Relationships

- **Correlation Heatmap:** Plotted to identify relationships between numeric features (e.g., the correlation between RAM and Price).

- **Price Distribution:** A histogram revealed the spread of computer prices, allowing us to check for skewness.

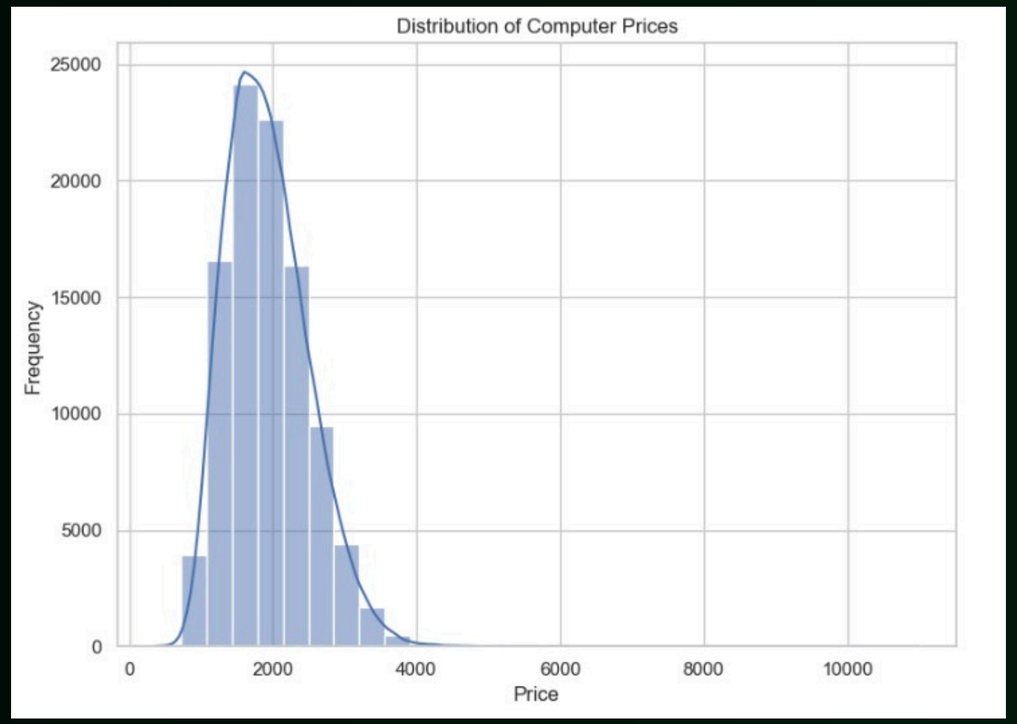
- **Pair Plots:** Visualized the interaction between key specs (cpu_base_ghz, ram_gb) against price.

- **Box Plots:** Used to detect outliers in hardware specifications like RAM.



EDA

BRAND	PRICE
Apple	2362.30
Razer	2079.53
Samsung	1930.39
MSI	1905.56
Dell	1882.82
Gigabyte	1866.30
Lenovo	1865.95
HP	1857.35
ASUS	1848.11
Acer	1760.35



Title: Price Distribution by Brand

We analyzed the average price across different manufacturers.

- **Premium Segment:** Apple has the highest average price, followed by Razer.
- **Mid-Range:** Samsung, MSI, and Dell .
- **Budget Friendly:** Acer had the lowest average price in this dataset.

EDA

mean absolute error (MAE):	$228.26
Standart deviation Absolute error:	$191.10
Absolute Residual:	$514.92
Number of the outliers:	1352

:WIFI
F-statistic 0.444 | p-value 0.7211
Result: there is no effect on price

Outlier Analysis

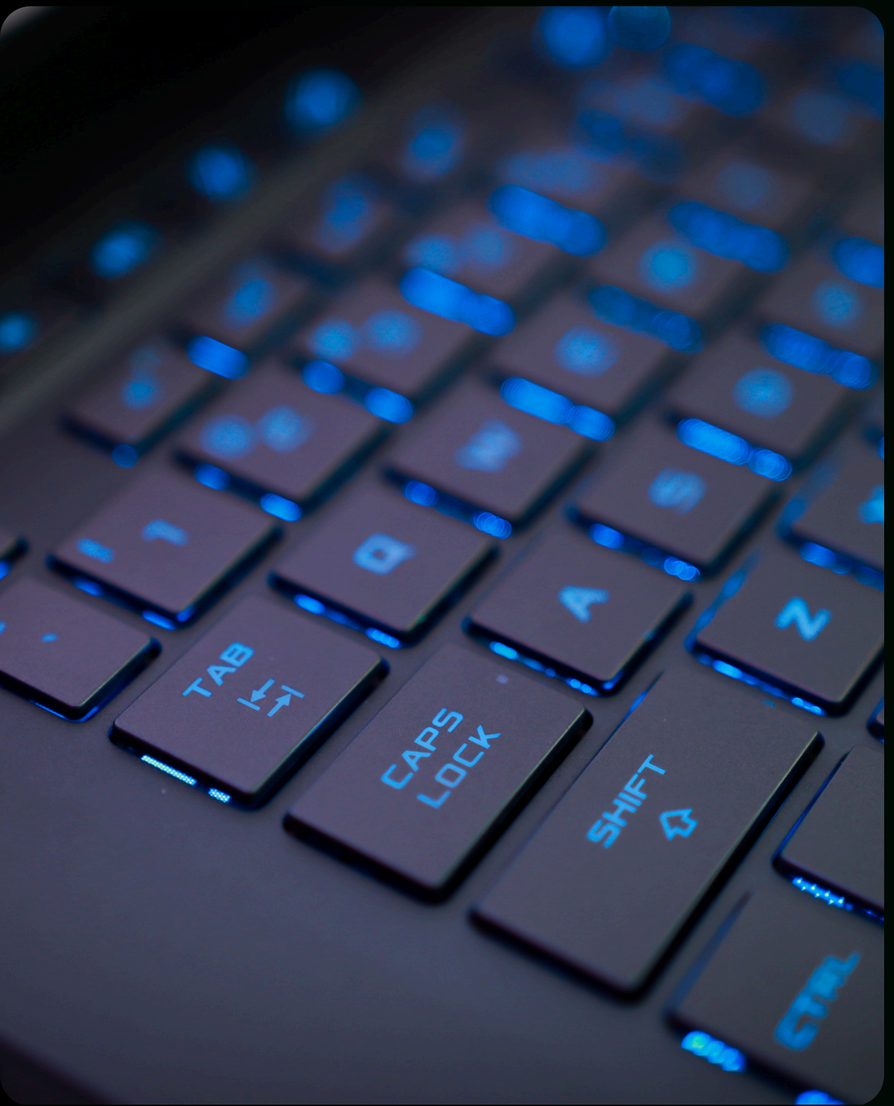
The model's prediction errors exhibit a degree of variability.

Additionally, the presence of a substantial number of outliers indicates notable deviations that may affect the model's predictive performance.

Which Categorical Features Matter?

We performed ANOVA (Analysis of Variance) tests to see if categorical features impact price.

WIFI: The P-Value was 0.72, indicating that the type of WIFI card has almost no effect on the final computer price.



EDA

Conclusion:

•Hardware specs (RAM, CPU, Storage) are strong predictors of price.

•Brand perception (e.g., Apple vs. Acer) plays a major role.

•Future Improvement: To improve accuracy on high-end machines (outliers), we should explore non-linear models like Random Forest, XGBoost, CatBoost to capture premium pricing logic better.

Price Prediction - *Dataset Preparation*

1. Data Cleaning & Preparation:

- Cleaned extreme outliers using IQR.
- Normalized skewed prices using log-transform for better training performance.
- Corrected GPU model names into the standard naming. For example, corrected RX 7000 80 XT to RX 7800 XT.

2. Feature Engineering:

- Extracted ppi (pixel per inch), screen width, screen height from resolution (e.g., “1920x1080") and display size.
- Extracted family, generation, tier and premium labels from CPU and GPU models, then combined into a benchmark score which will be explained later in slide.
- Categorical columns are encoded with One Hot Encoding to improve model learning.
- In our dataset, CPU and GPU models were weakly correlated with price, so we benchmarked each component and reassigned models based on realistic performance tiers.

	release_year	vram_gb	ram_gb	storage_gb	display_size_in	refresh_hz	price	screen_width	screen_height	ppi	cpu_gen	cpu_tier	gpu_gen	gpu_tier	gpu_premium	cpu_score	gpu_score	brand_ASUS	brand_Acer	brand_Apple	brand_Dell
0	2022	6	16	1024	27	90	7,232726	2560	1440	108,786	11	129	4	60	0	579,234	2200	0	0	0	0
1	2022	10	64	512	16	90	7,729731	1920	1080	137,682	11	114	4	80	0	782,928	3300	0	0	0	0
2	2024	4	8	512	32	120	7,539022	3440	1440	116,539	5	168	4	50	0	654,303	1540	0	0	0	0
3	2024	6	16	512	27	120	7,194429	3440	1440	138,12	7	550	7	60	0	908,86	1710	0	0	0	1
4	2024	12	96	256	15,6	90	7,894314	2560	1600	193,518	6	230	3	80	1	943,143	2760	0	0	0	0
5	2025	16	96	512	24	90	7,92008	2560	1440	122,384	10	369	4	90	0	847,748	3960	0	0	0	0
6	2024	0	8	2048	32	60	7,383983	2560	1440	91,7878	2	1	1	1	0	966,046	280	0	0	1	0
7	2023	0	32	1024	27	60	7,668556	2560	1440	108,786	2	2	1	1	0	1104,184	280	0	0	1	0
8	2024	16	128	1024	14	60	7,990912	2560	1600	215,634	14	473	4	90	0	1307,267	3960	0	0	0	1
9	2025	4	8	512	15,6	120	7,410946	3840	2160	282,424	4	374	3	50	0	499,75	1120	0	0	0	0

Price Prediction - *How the System Works*

3. Splitting & Preprocessing:

- The data is split into 80% train and %20 test dataframes.
- Numeric and categorical columns are defined for preprocessing. For the Stacking model, categorical features are additionally transformed using CatBoostEncoder inside a preprocessing pipeline.

4. Training Process:

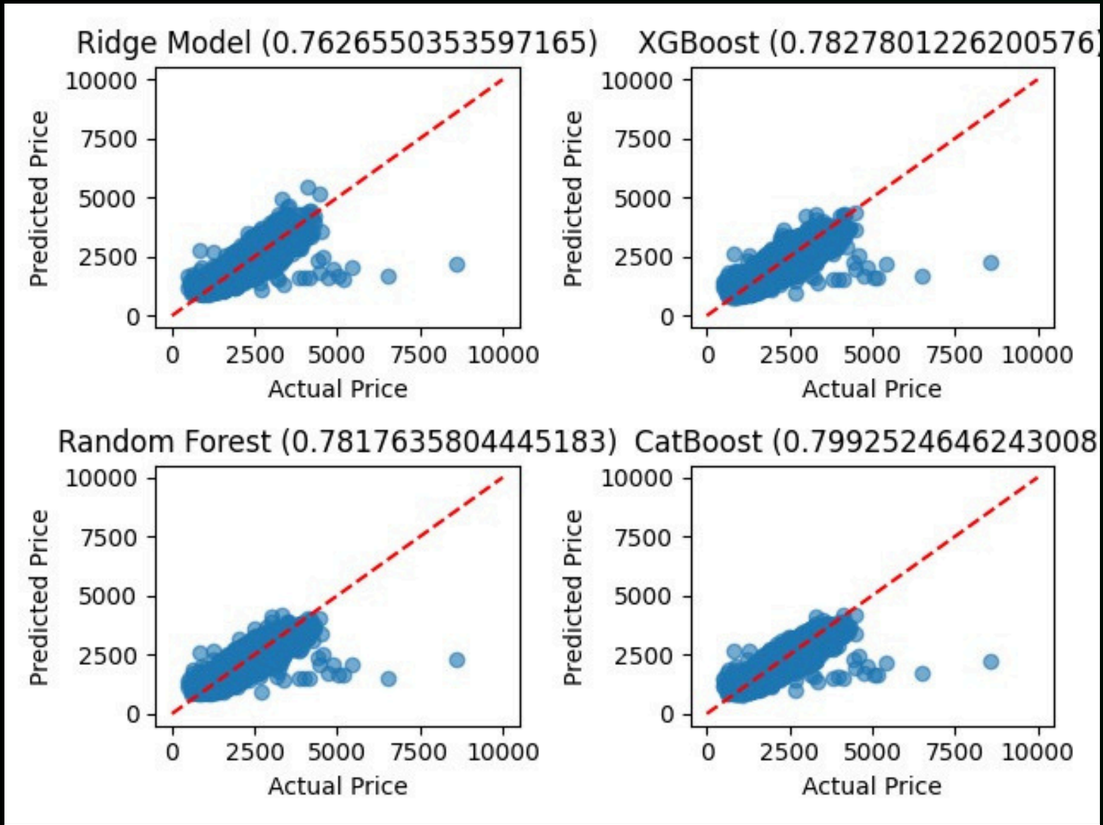
- We trained 4 models in this task. Random Forest Regressor, XGB Regressor, CatBoostRegressor and StackingRegressor which is combined version of previous models (RF + LGBM + CatBoost + XGB).
- Pipeline is used to connect preprocessing and model training process. Linear Regression is used as the final estimator to combine model outputs in a simple and stable way, reducing overall model variance.
- GridSearchCV used for tuning Random Forest hyperparameters which will help finding optimal parameters.
- After training, all models were evaluated and the best one was exported for use in the price prediction interface.

```
--- Random Forest ---
R-squared (R²): 0.9288
MAE (Mean Absolute Error): $114.85
RMSE (Root Mean Squared Error): $150.73

--- CatBoost ---
R-squared (R²): 0.9353
MAE (Mean Absolute Error): $109.54
RMSE (Root Mean Squared Error): $143.62
```

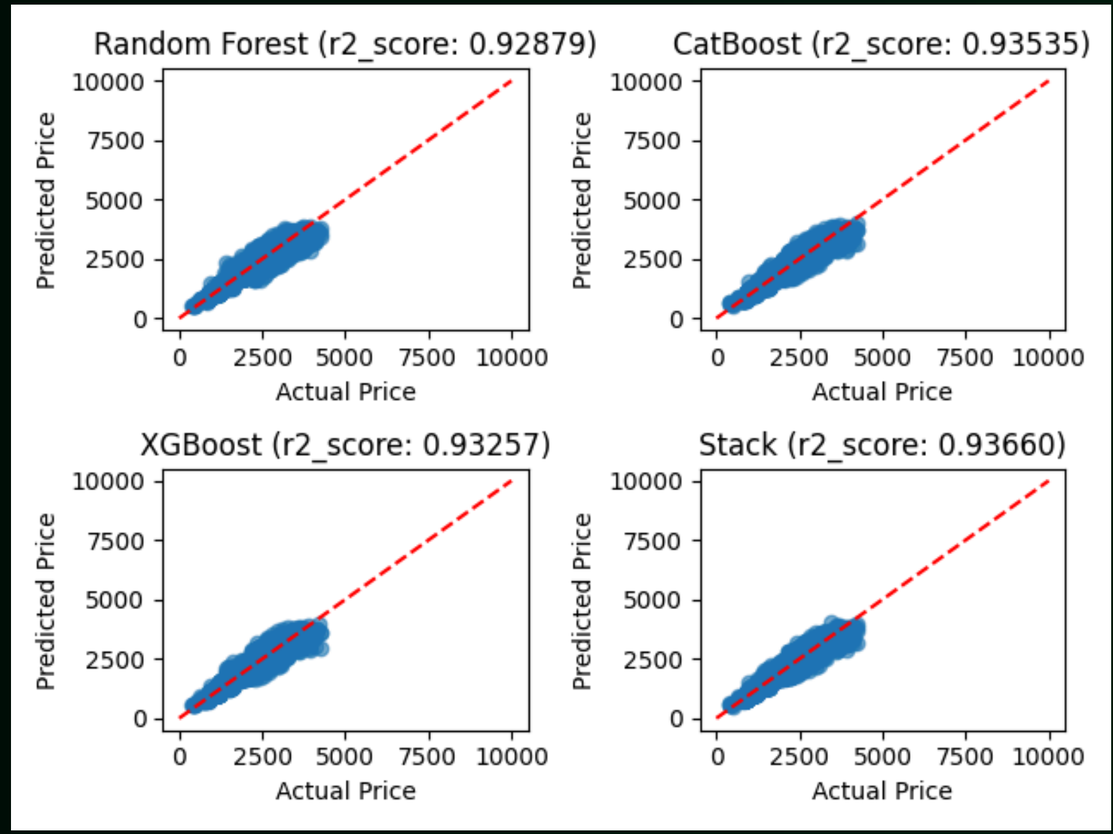
```
--- XGBoost ---
R-squared (R²): 0.9326
MAE (Mean Absolute Error): $111.25
RMSE (Root Mean Squared Error): $146.68

--- Stack ---
R-squared (R²): 0.9366
MAE (Mean Absolute Error): $108.67
RMSE (Root Mean Squared Error): $142.23
```

Price Prediction – *Before & After Feature Engineering*

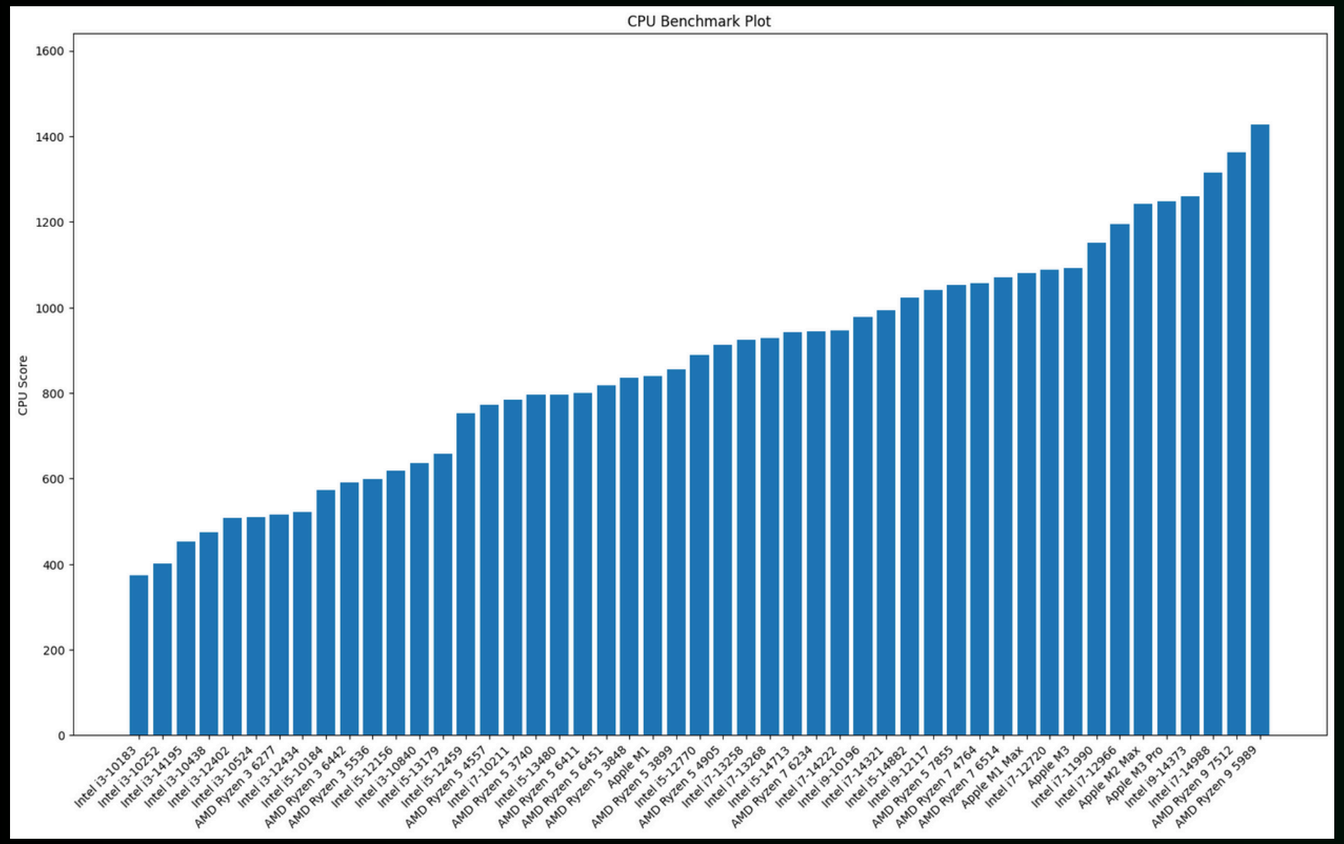
BEFORE



AFTER



Benchmark Calculation - *CPU Score*



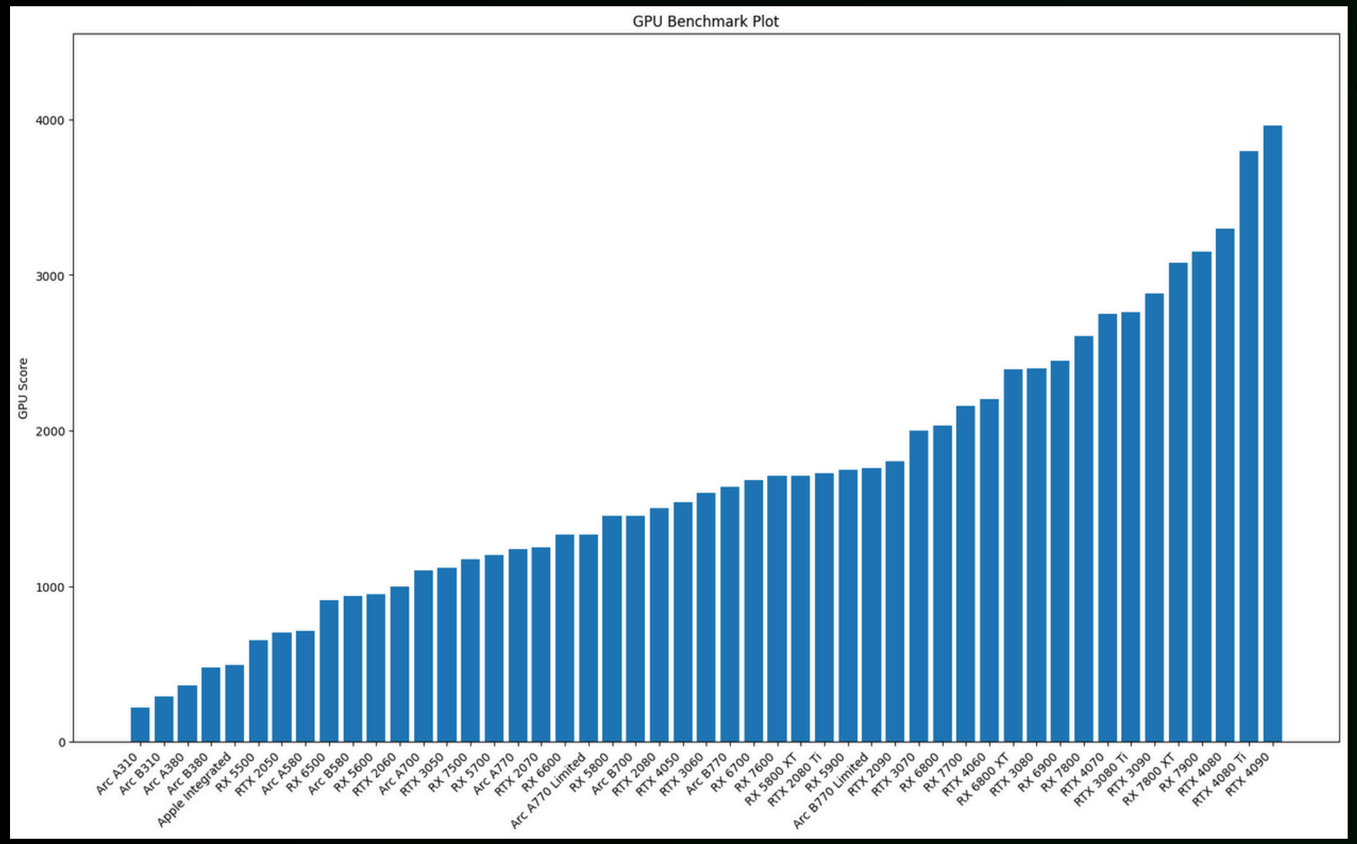
Our CPU score is a unified numeric metric designed to compare processors across brands.

- **Composite Scoring Formula:** We combine family, generation, and tier information into a single unified score.
- **CPU Family:** Each CPU family is mapped to a performance tier. We classify the processor into a performance family using pattern matching.
- **Generation:** We detect the processor generation by taking first digits of model number.
- **Tier:** Each CPU tier is calculated by their remaining digits.

Examples: Intel i9-14373 AMD Ryzen 9 7512 Apple M3 Pro

CPU Family Generation Tier

Benchmark Calculation - *GPU Score*



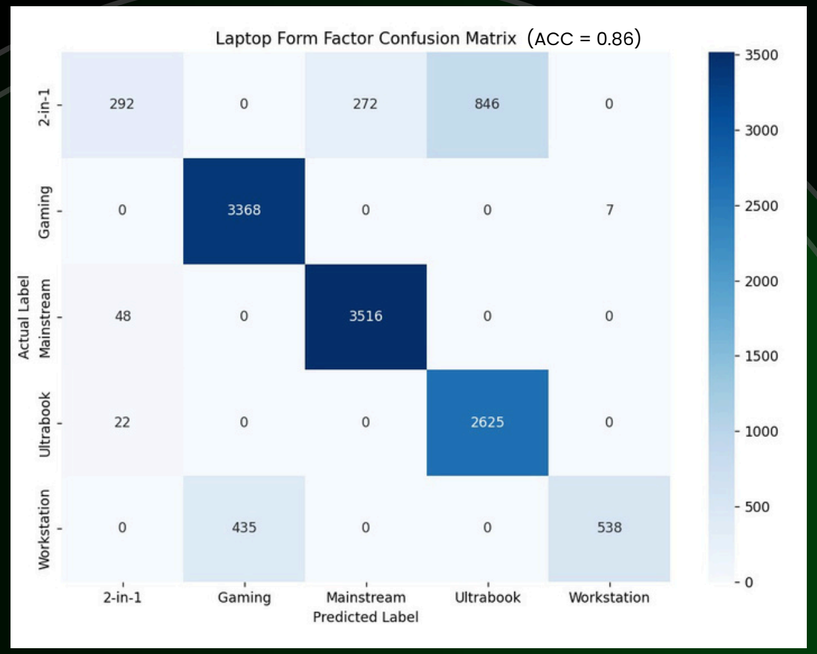
GPU score calculation is similar to CPU score calculation except a few differences.

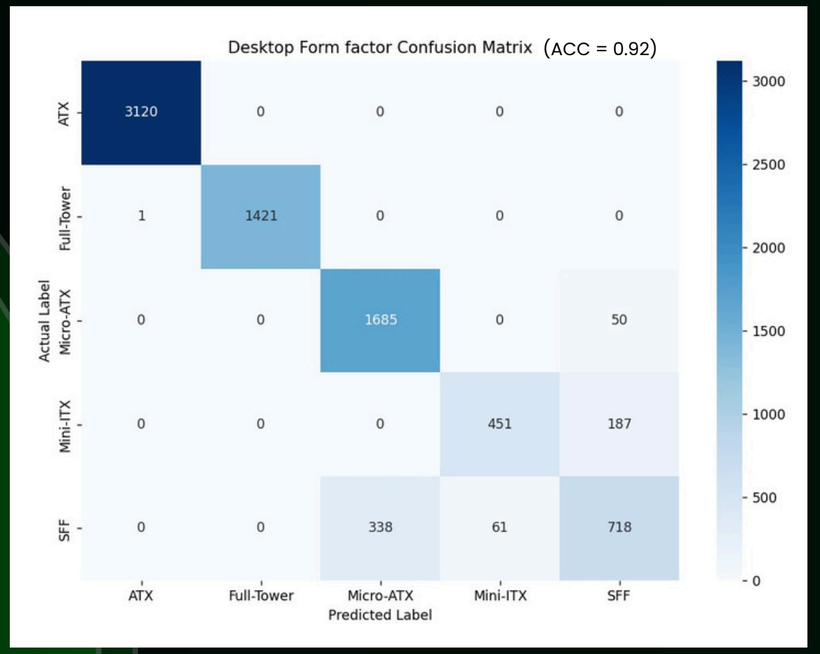
- **Generation:** Architectural generation is extracted and incorporated into the score to reflect technological advancement.
- **Tier:** Model naming conventions are mapped into performance tiers representing expected capability levels.
- **Premium Adjustments:** Higher-end models receive tier adjustments to account for known performance jumps.

Examples: Arc B770 Limited RX 7800 XT RTX 4080 Ti

Generation Tier Premium Label

Device Classification





Feature	Importance
weight_kg	0.575859
gpu_tier	0.036990
battery_wh	0.028600
display_type	0.025539
charger_watts	0.025391
cpu_boost_ghz	0.024302
bluetooth	0.023144
storage_gb	0.022139
refresh_hz	0.021568
display_size_in	0.021334
wifi	0.019320
storage_type	0.019301
cpu_threads	0.18470
warranty_months	0.018139
vram_gb	0.017492
resolution	0.016983
ram_gb	0.016592
os	0.013813
cpu_cores	0.012391
gpu_brand	0.012321
cpu_brand	0.010300
storage_drive_count	0.007056
cpu_tier	0.006657
cpu_base_ghz	0.006301

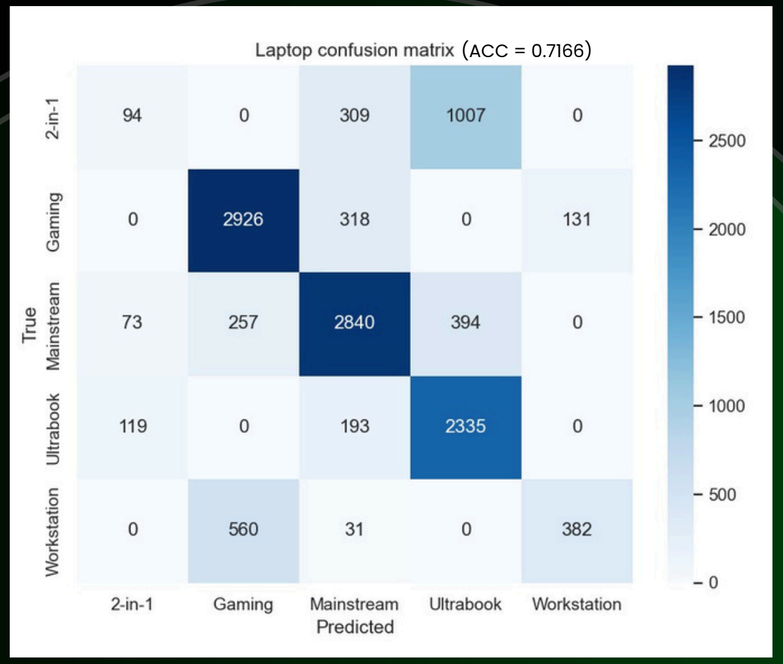
First Model's Areas of Improvement

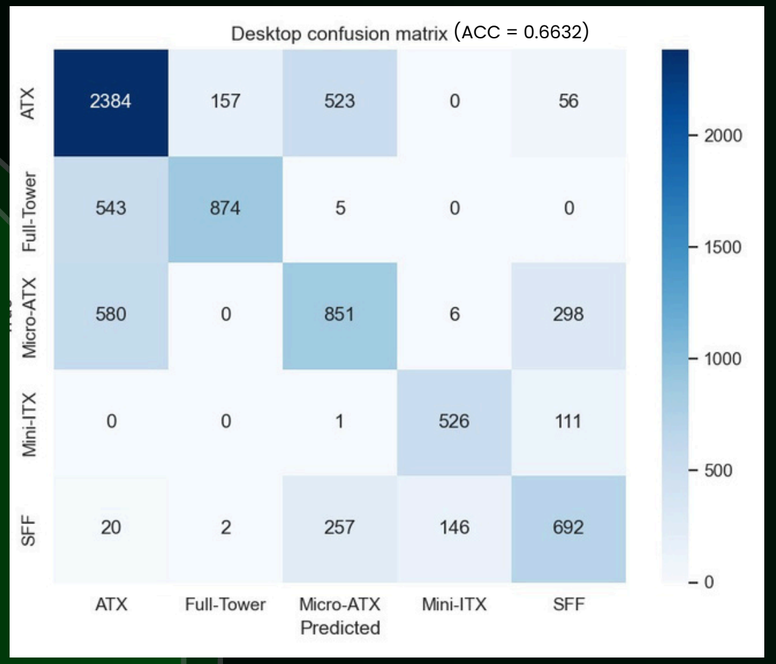
The model struggled significantly with specific niche categories in the Laptop segment:Difficulty in Identification: The classifier showed low Recall for 2-in-1 Recall = 0.23 and Workstation Recall = 0.56 laptops.

Model Strength

The model achieved near-perfect scores for ATX F1-Score = 1.00 and Full-Tower F1-Score = 1.00 desktop cases.Insight: The distinct physical attributes primarily the weight and PSU wattage associated with these large form factors provided the model with a clear, unambiguous signal.

Device Classification





Feature	Impotance
weight_log	0.198210
weight_kg	0.162218
portability_index	0.131845
density_per_core	0.082226
gpu_density	0.071948
orig_weight_kg	0.024431
psu_watts	0.022195
cpu_boost_ghz	0.020650
display_type	0.020092
bluetooth	0.018086
cpu_threads	0.017721
refresh_hz	0.017660

Second Model's Areas of Improvement

Difficulty in Identification: The classifier showed low Recall for 2-in-1 Recall = 0.17 and Workstation Recall = 0.51 laptops. In the Desktop classification, the model performed poorly on the Micro-ATX class (F1-Score = 0.50), showing difficulty in separating mid-sized, similar form factors.

Model Streght

The model achieved excellent separation for the largest and most distinct form factors across both categories

Laptops: Achieved high F1-Scores for Gaming (0.81) and Mainstream (0.78).

Desktops: Showed strong performance for Mini-ITX F1-Score = 0.80, ATX F1-Score = 0.72, and Full-Tower F1-Score = 0.71 cases.

Insight: The high accuracy in these categories suggests that the distinct physical attributes and specifications associated with these form factors provided the model with a clear, unambiguous signal, allowing for accurate classification.

Device Classification

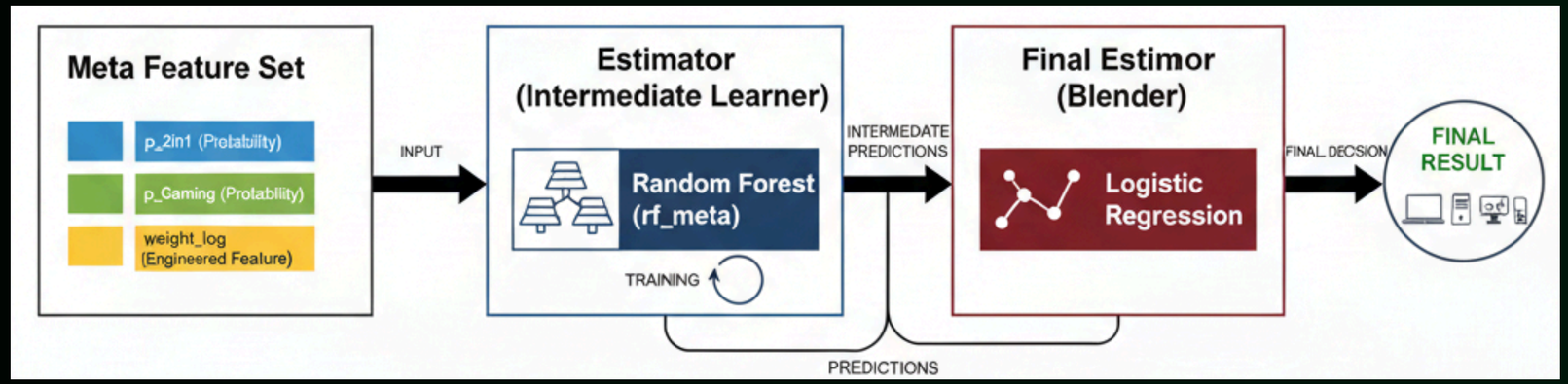
Final Model (Hierarchical Stacking)

Why Hierarchical model : Form Factors represent highly distinct sub-categories within the broader Laptop/Desktop types. The data is often imbalanced. Using device-specific sub-models provides greater precision than attempting a single large multi-class prediction.

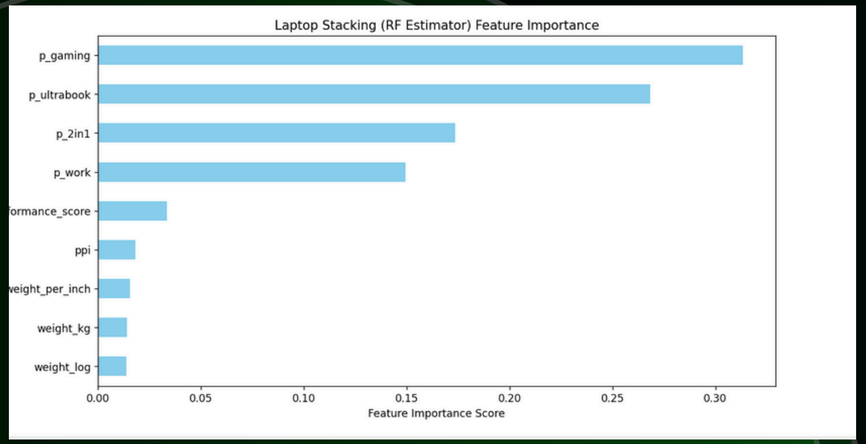
Base Learners : We trained specialized Random Forest models to distinguish each specific form factor against all others. Oversampling was critically used for minority classes.

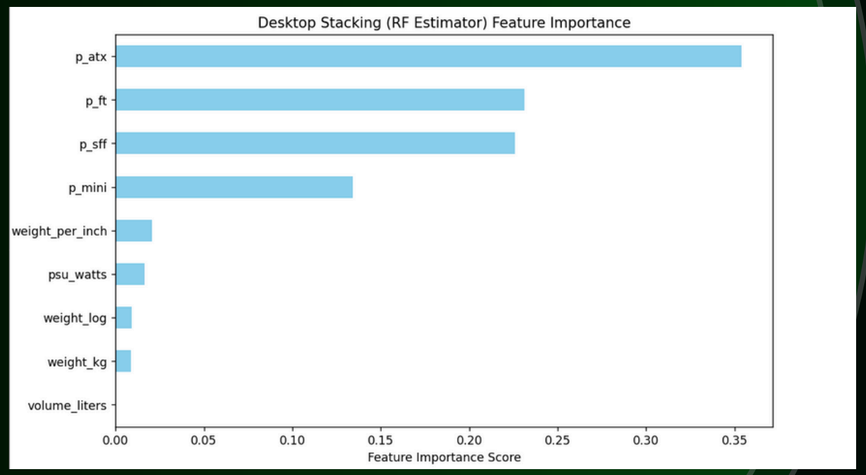
Meta Feature : The trained binary models outputted probability scores for each device. These probability scores are collected and transformed into a new set of features the Meta Features. This step explicitly transfers the "confidence" of the base models to the final model.

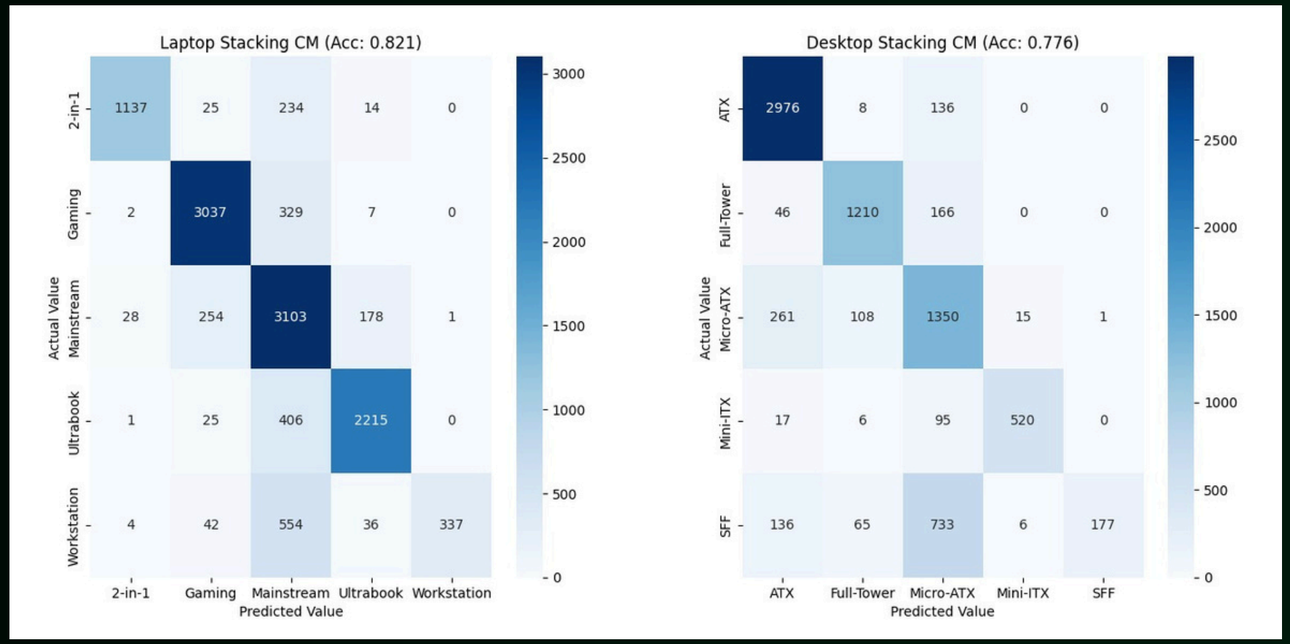
Stacking: The Meta-Features were concatenated with critical engineered features . A StackingClassifier was trained on this composite feature set to perform the final multi-class Form Factor prediction.



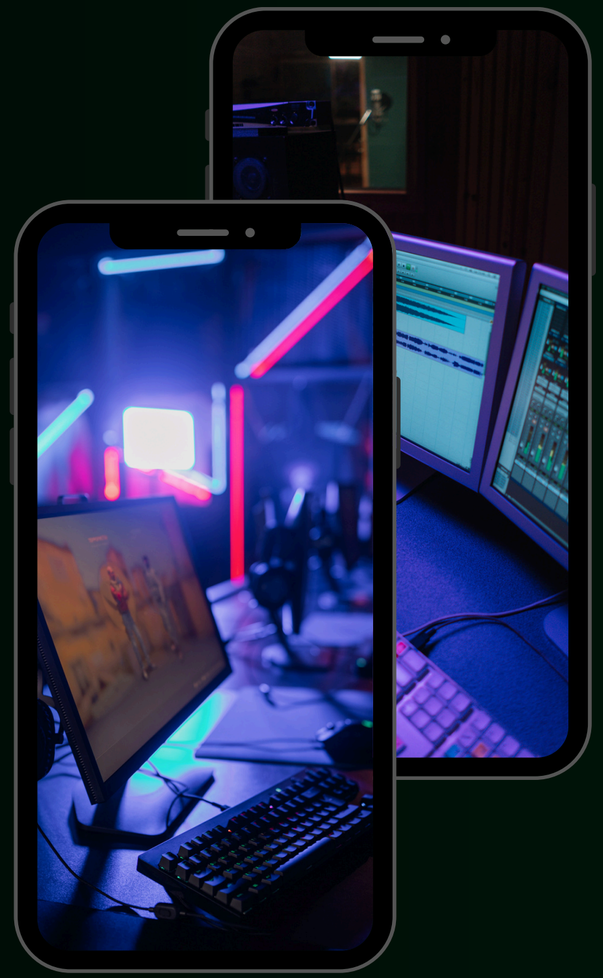
Device Classification







Show the Project



END OF PRESENTATION

THANKS FOR LISTENING TO US :)