Predicting User Experience on Laptops from Hardware Specifications

Saswat Padhi — spadhi@
Sunil Bhasin — skbhasin@
Udaya Kiran Ammu — udaykiran@
Alex Bergman — abergman@
Allan Knies — aknies@

Summary

- **Problem:** Accurately predicting everyday end-user experience (UX) on laptops

  GeekBench, SPEC etc. measure the "peak" performance of subsystems, but not the "average" system UX on everyday tasks.

- **Scope:** Chromebook laptops, which mainly run web applications on ChromeOS
- 100 K data points from 54 Chromebooks on 9 UX metrics from Chrome browser
- Regression models are trained to predict UX metric values from hardware specs

Methodology

- Train one regression model per UX metric to estimate that metric's value from a laptop's hardware specifications

Estimated Metric Value — Regression Model — Model Parameters

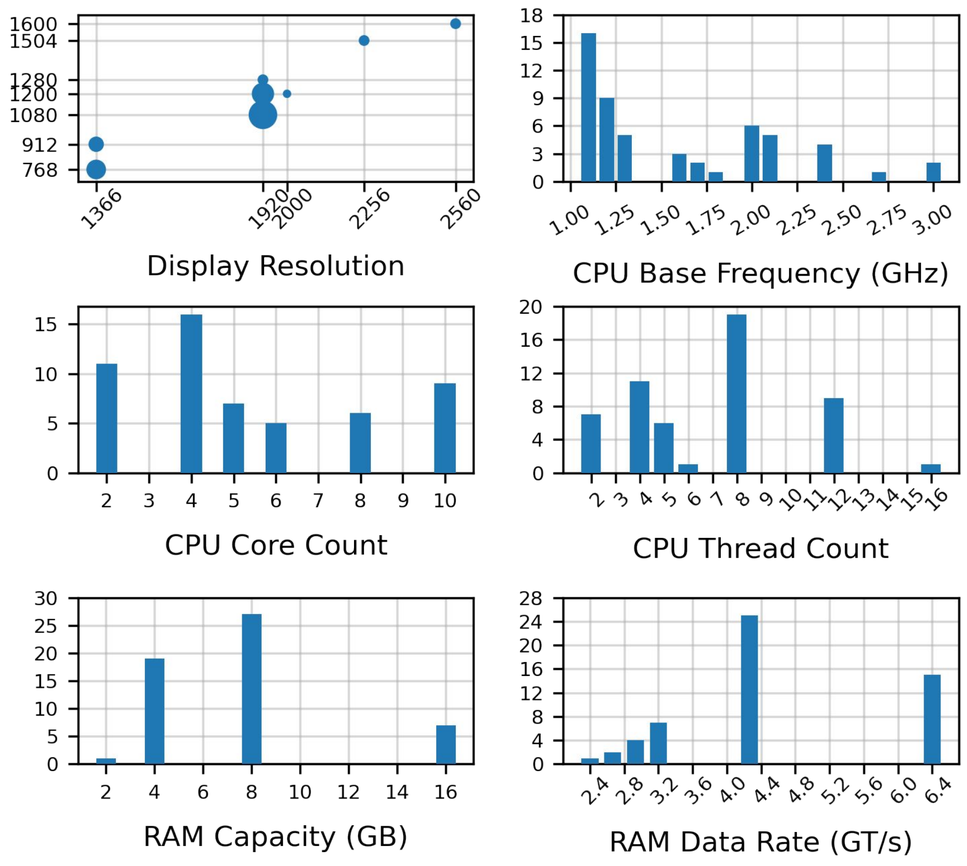
$$\hat{y}_m = f_m(\mathbf{x}, \hat{\beta})$$

where

$$\mathbf{x} = \langle x_{\text{cpu\_freq}}, \ldots, x_{\text{ram\_capacity}}, \ldots, x_{\text{display\_res}} \rangle$$

- Gradient Boosted Regression Trees (GBRTs)
  Mean Squared Error (MSE) loss function
  Grid search to optimize hyperparameters

Data Collection



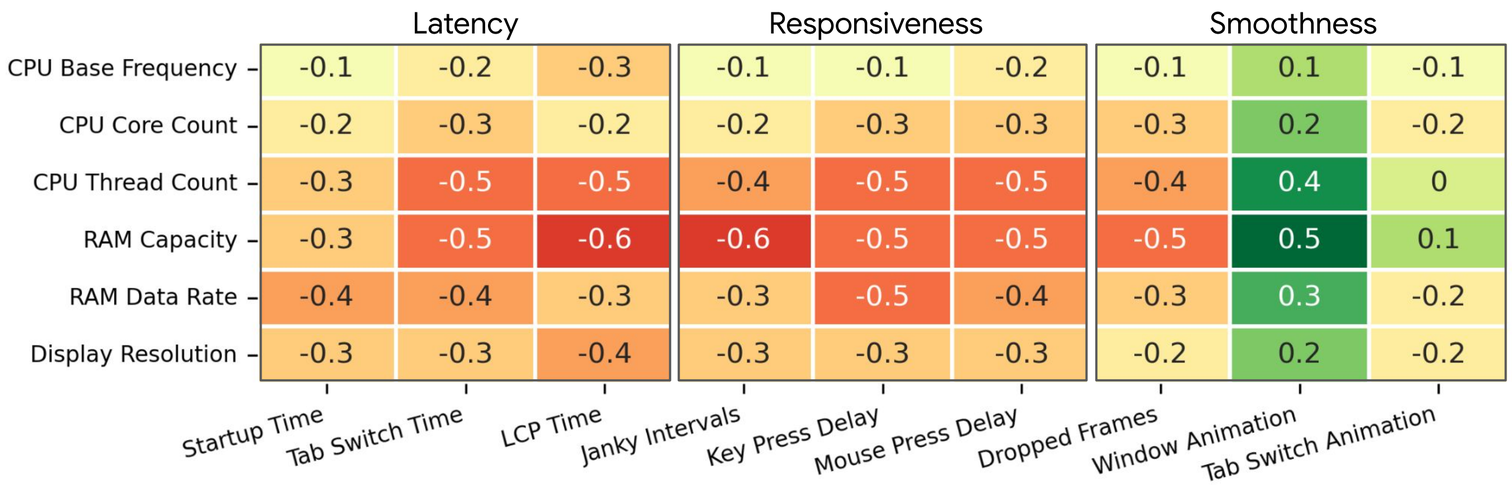
- Feature vector includes one-hot encoded System-on-Chip (SoC) vendor name
- Display resolution reduced to an integer: pixel count = horizontal × vertical pixels
- Automated tests for UX metric collection mimic typical end-user tasks, including web browsing, document editing, audio/video playback when Chromebooks are on AC power

UX Metrics

Latency	Responsiveness	Smoothness
Startup Time Time (ms) since app invocation to window launch	**Janky Intervals** Number of 100ms intervals in which a user event was waiting in queue	**Dropped Frames** Fraction (%) of frames dropped during scrolling or update
Tab Switch Time Time (ms) since a tab switch event to the first rendered frame	**Key Press Delay** Time (ms) taken by an app to start handling a key press event	**Window Animation** Relative (%) FPS (compared to 60) during window hiding animation
Largest Contentful Paint Time (ms) taken to paint the largest image or text block	**Mouse Press Delay** Time (ms) taken by an app to start handling a mouse press event	**Tab Switch Animation** Relative (%) FPS (compared to 60) during tab switching animation

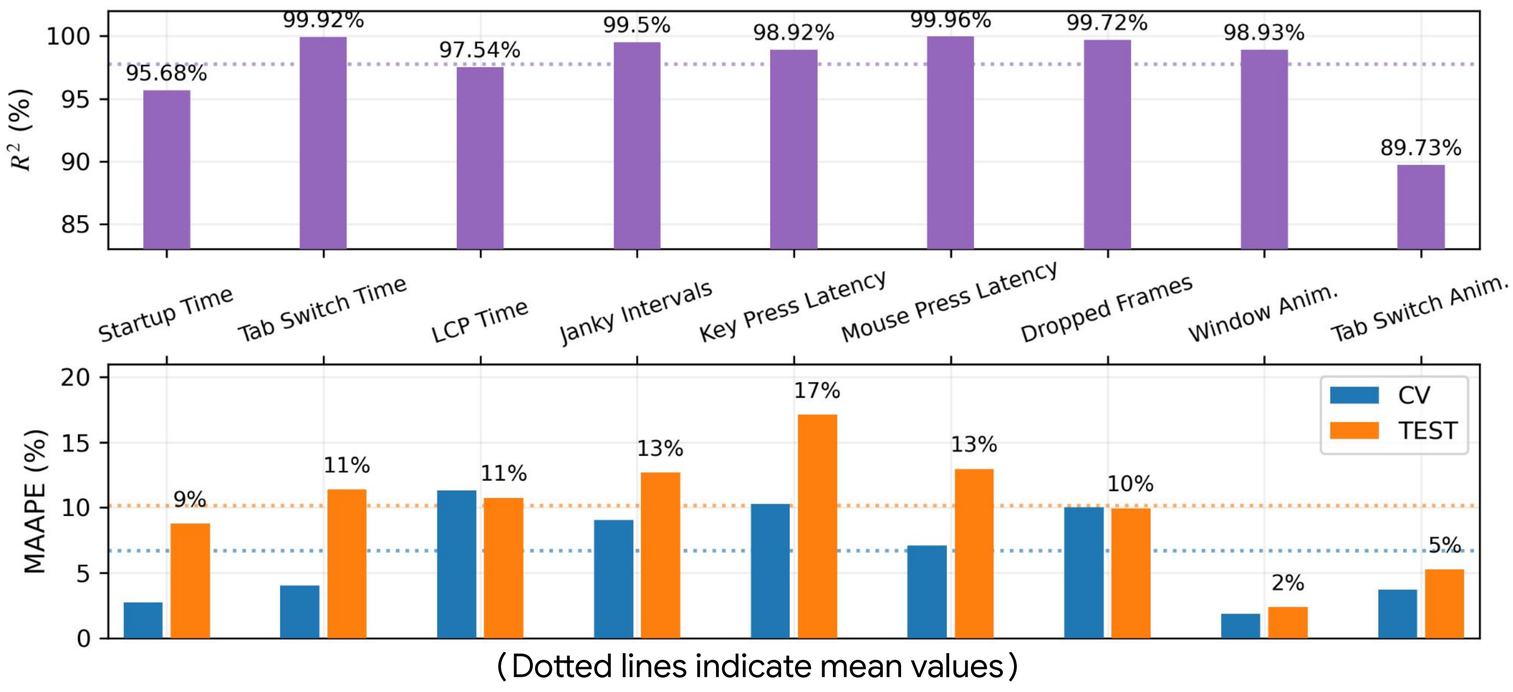
Key Findings

Kendall's Rank (τ) Correlation on Hardware Specs & UX Metrics:



CPU thread count and RAM capacity show strongest correlation with better UX
Latency and frame drop correlate **negatively** Animation smoothness correlate **positively**

R^2 fits and MAAPE errors of GBRT predictors:



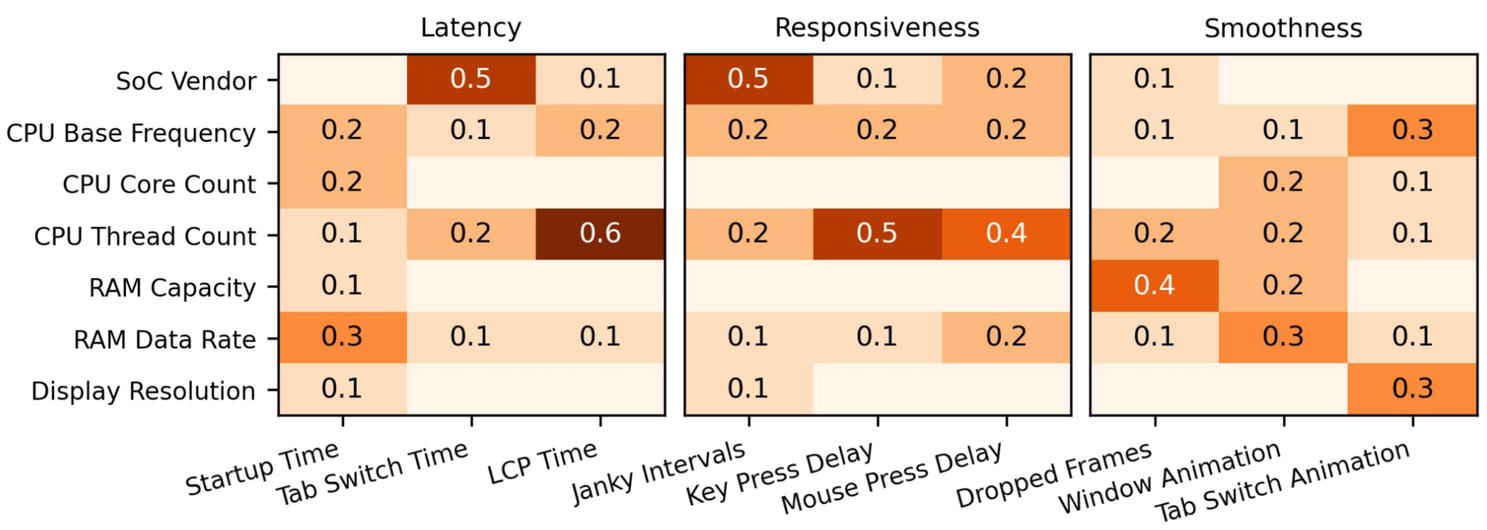
(Dotted lines indicate mean values)

High R^2 ⇒ Models capture data variance well ("in-sample": on training dataset)

Low MAAPE ⇒ Models predict accurately ("out-of-sample": on test & CV datasets)

MAAPE provides a stable relative error, even when the true values are zero

Normalized Permutation Feature Importance for GBRTs:



(White cells denote feature importance below 0.05)

CPU thread count has **high importance** across all predictors
Many predictors are vendor-agnostic: generalize purely on specification numbers