NOT ALL MOBILE DEVICES ARE EQUAL: MEASURING HETEROGENEOUS PERFORMANCE IN ONLINE CONTROLLED EXPERIMENTS

Evan Chow
Snap, Inc.
echow@snap.com

Yuxiang Xie
Snap, Inc.
yxie@snap.com

Xiaolin Shi
Snap, Inc.
xiaolin@snap.com

1 Abstract

It is of great importance to accurately measure various performance features in the production cycle of mobile apps in industry. Unlike measuring user engagement, whose metrics are usually aggregated on the user level (i.e. one data point per user) in A/B experiments, the industry standard is to calculate performance metrics by aggregating on the event level (i.e. one data point per event). However, since the distributions of event count from each device (or user) are usually extremely skewed, event-level performance metrics can be dominated by a small group of heavy users on relatively high-end mobile devices. Thus, optimizing for event-level performance metrics in A/B testing experiments can bias product decisions toward the experiences of heavy users. On the other hand, if we measure performance metrics in the same way as user engagement metrics, the experiences of all users can be weighted equally. We compare both the event-level and user-level approaches to measuring performance for 152 A/B experiments at Snapchat, and find that 10.06% of results show differences in statistical significance, effect direction, or both. We further show that it is the strong heterogeneity of mobile devices on the market that causes the non-trivial gap between event-level and user-level performance metrics. Based on our investigation of this device heterogeneity, we provide recommendations on how to measure and interpret performance metrics for A/B experiments of mobile apps and discuss directions for future work.

***K*eywords** mobile computing · heterogeneous treatment effects · performance latency · digital experimentation

2 Introduction

Long app loading times on mobile devices hurt user experiences [5, 10, 13], drive away user traffic [1], lower revenue [6], and cause energy drain [3]. To measure these latencies, the current industry standard is to compute a quantile, such as the median (the 50th) across all performance events uploaded from all mobile devices and traffic sources [19, 20]. We denote this state-of-the-art measurement approach, which is discussed in [14], as *event-level performance metrics*. Quantiles are preferred to averages for robustness to outliers, and to this end industry has historically tracked both mid-tier quantiles (50th, 75th) as well as worst-case tail quantiles (90th, 99th) [4, 9, 17].

However, since the distributions of event count from each device (or user) are usually extremely skewed, metrics calculated by this industry-standard approach to measuring mobile performances are usually dominated by a very small group of heavily engaged users with faster and higher-end mobile devices, while leaving the mobile performances of non-heavy users underrepresented. At Snapchat, we find the former produce disproportionately more performance events and are thus overrepresented by event-level metrics. In our first example experiment, users in the top 10% of performance emit approximately 2.5 times as many performance events as those in bottom 10%. On the other hand, performance varies widely across mobile devices [8, 16], and even having the same model of phone does not guarantee identical performance [2]. Due to this heterogeneity, optimizing for event-level metrics in A/B experiments can mislead stakeholders to engineer performance improvements only for the small group of heavy users with limited types of mobile devices, at the expense of the less-engaged majority.

To deal with this, we propose *user-level performance metrics* which aggregate all events at the user-level first. Event-level metrics safeguard a company's heavy users, while user-level metrics track the impacts across the full user spectrum. In this work, we empirically compare event-level and user-level performance metrics on 152 real-world A/B tests at Snapchat and show that, due to the heterogeneity of mobile devices, different measurement approaches toward performance can lead to dramatically different observations in A/B experiment results. We present two example experiments that benefit or preserve the performance of heavy users while hurting the performance of non-heavy ones. Finally, we discuss our continued progress and summarize takeaways.

3 Event-level vs. user-level metrics

Suppose we are interested in a particular performance metric Ω, such as time to load Snapchat. In a given A/B experiment, each user i has an individual set of K_i observations of that metric $\Omega_i = \{\omega_{ij} | 1 \leq j \leq K_i, \omega_{ij} \in \mathbb{R}, \omega_{ij} > 0\}$. Let the quantile function for probability $p \in (0, 1))$, over some set of real-valued observations Ω, be denoted as $Q^p(\Omega) = \inf\{\omega \in \Omega | p \leq F(\omega)\}$ where $F(.)$ is a continuous, strictly monotonic cumulative distribution function. We consider two possible measurements of the same metric Ω which we denote as the *event-level metric* and the *user-level metric*.

Event-level metric Ω: For a set of N users compute the desired quantile directly over all events collected during the experiment, given by $\Omega^p(\Omega) = \inf\{\omega \in \Omega | p \leq F(\omega)\}$ where $\Omega = (\omega_1 \cup \omega_2 \cup \ldots \cup \omega_N)$. Do this separately for treatment and control users, calculate quantile variance [18], then proceed as usual with A/B experimentation [12]. This method is straightforward to implement and is common in industry, but overweights heavy traffic sources and is not amenable to diagnosing device heterogeneity.

User-level metric Ω: For a set of N users, compute the desired quantile per each user in the experiment. Then, take the mean μ^p over all per-user quantiles. This is given by $\mu^p = \frac{1}{N} \sum_{i=1}^{N} Q^p(\Omega_i)$ where $Q^p(\Omega_i) = \inf\{\omega_{ij} \in \Omega_i | 1 \leq j \leq K_i, \omega_{ij} \in \mathbb{R}, \omega_{ij} > 0, p \leq F(\omega_{ij})\}$. Do this separately for treatment and control users, then proceed as usual with A/B experimentation [12].[1] This method weights all users equally and is more comparable with user-level mobile engagement metrics such as Average App Open Per User. However, it is less straightforward to implement, and quantile calculations are not accurate when users have few events.

Event-level metrics primarily safeguard the experiences of heavy users on fast devices, while user-level metrics track the full range of heavy and non-heavy users more equitably. Thus, using both in tandem allows us to better detect heterogeneous impacts across many kinds of devices and users.

4 Empirical Findings

We examine 152 A/B experiments from August 2019 in order to understand how performance metrics differ between the event-level and the user-level due to device heterogeneity. These experiments span app improvements, product and design changes, etc. We only consider experiments that have sufficiently large sample size and do not show sample ratio mismatch in user count [7]. For each experiment we examine up to 5 particular choices of performance metric Ω. For each performance metric we examine 11 particular choices of quantile Q^p with $p = 0.1, 0.2, \ldots, 0.9$, $p = 0.01$ and $p = 0.99$. This totals to 8,360 results, but we filter out those that show sample ratio mismatch, therefore yielding our final 8,338 individual performance quantile results across the 152 experiments. For each performance quantile result, we examine differences between its event-level metric and its user-level metric.

Our first key finding is that A/B results can display differences in significance, effect direction, or both depending on whether the same performance quantile is measured at the event-level or the user-level. Table 1 breaks down the 8,338 total results by differences in significance ($p \leq 0.05$), effect direction (whether a result is positive or negative), or both. A result may be significantly positive, significantly negative, or insignificant (regardless of effect direction). Results are tabulated by absolute count and % of total.

Event vs. User	**Positive, significant**	**Negative, significant**	**Insignificant**
Positive, significant	39 (0.47%)	**1 (0.01%)**	**178 (2.13%)**
Negative, significant	**12 (0.14%)**	41 (0.49%)	**214 (2.57%)**
Insignificant	**213 (2.55%)**	**221 (2.65%)**	7419 (88.98%)

Table 1: *How frequently results change when measured at the event-level (rows) vs. at the user-level (columns), for 8,338 total performance metrics in 152 experiments.* ***10.06%*** *change significance (*$p \leq 0.05$*), effect direction, or both.*

We find approximately 10.06% of all performance metrics show differences in significance, effect direction, or both between event-level and user-level. It is most common for insignificant results to become significant and vice-versa. The remaining 89.94% along the left diagonal do not show differences, which is expected since most experiments may not focus specifically on performance. We reemphasize that these disparities between event-level and user-level metrics are caused by the heterogeneity of mobile devices. Heavily engaged users on fast devices contribute more events than non-heavy users on slow ones, and therefore become overrepresented in event-level metrics. In contrast, user-level metrics weight all users equally and thus represent the full spectrum of performance.

[1] We do not use the median to aggregate, since the mean is better supported in industry experimentation.

Experiment 1: We present an experiment that shows substantial and significant improvements (speed boost) to the P50 of a certain performance metric when calculated at the event-level, as shown in Table 2. However, when calculated at the user-level, the impact becomes insignificant. This discrepancy suggests heavily engaged users with faster devices may be overrepresented here on the event-level. To investigate this, we run quantile regression on a sample of 500,000 users from the experiment (Figure 1), similar to [15]. Essentially, to understand impact over different parts of the performance spectrum, we compare performance under the same user-level percentiles between treatment and control. This is represented on the x-axis, moving from users on slow devices (left) to users on fast devices (right). For instance, P40 indicates we compare the 40th-percentile user in treatment to the 40th-percentile user in control. This comparison yields the percentage treatment effect denoted on the y-axis, where a positive value indicates speed improvement.[2] We see that slow users (left) are mostly unaffected, while medium to fast users (right) see substantial speed improvements.

Investigating further we see that faster users, for each percentile along the x-axis, had increasingly higher average event counts (Figure 2). This demonstrates that the event-level metric result primarily represents faster and more heavily engaged users, which directly explains why it is substantially higher than the more equitable user-level one.

	Average Treatment Effect Speed improvement
Event-level metric	**+0.3120%** ($p \ll 0.001$)
User-level metric	+0.0816% ($p = 0.811$)

Table 2: *Differences in effect and statistical significance between event-level and user-level metrics, over the full experiment population.*

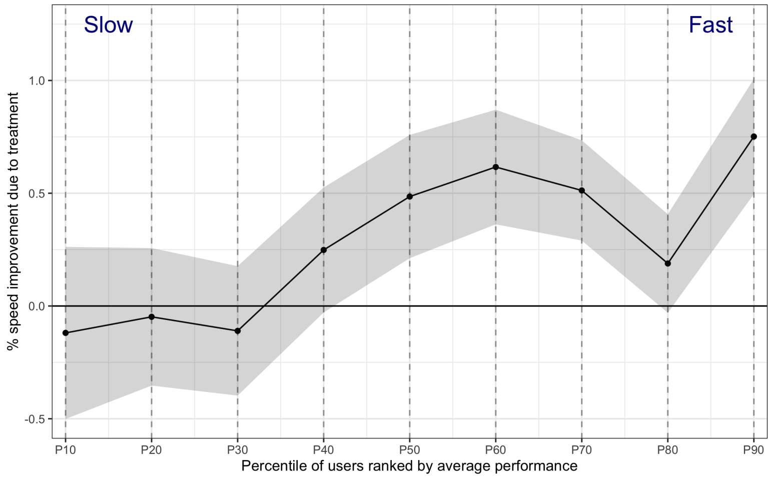

Figure 1: *The experiment impacts fast and slow devices differently. Slow devices see little or no change, while mid-range and fast devices see substantial speed improvements. Specifically, we use quantile regression to compare performance under the same user-level quantile. Each user-level observation is the 50th-percentile of a single user's observations. For instance, the point (P60, 0.63) represents a speed boost of +0.63% between the treatment 60th-percentile user and the control 60th-percentile user. Confidence intervals are calculated from a Huber sandwich estimate described in [11].*

[2] Since performance metrics measure latency (the inverse of speed), we flip our results to denote improvement to performance or speed. Thus, a positive effect indicates a beneficial improvement.

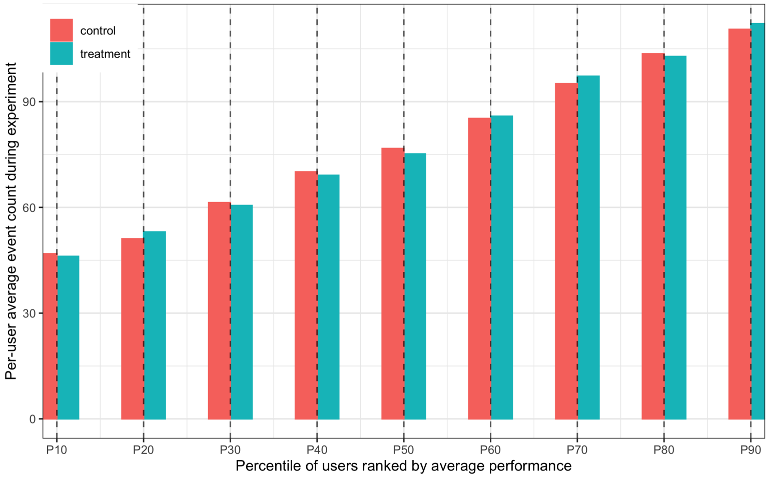

Figure 2: *Per-user average event count, bucketed for the performance percentiles above, for both treatment and control. For instance, the red bar (P30, 61) indicates the average user in the 30th-percentile of performance in control had 61 performance events during the experiment duration.*

Experiment 2: In our second case study experiment, the event-level metric is flat but the user-level one shows a significant negative impact, as seen in Table 3. Investigating this discrepancy with 125,000 users in Figure 3, we see that this experiment mostly preserves the performance of mid-range and fast users, but worsens the performance of slow users. Examining the distribution of event count in Figure 4, we see here that event-level metrics primarily reflect the contributions of heavier mid-range and fast users, thus explaining why the event-level metric is insignificant. Only the user-level metric captures the significant negative impact to slow users.

	Average Treatment Effect Speed improvement
Event-level metric	-0.3228% ($p = 0.8684$)
User-level metric	**-12.5717%** ($p = 0.0303$)

Table 3: *Differences in effect and statistical significance between event-level and user-level metrics, over the full experiment population.*

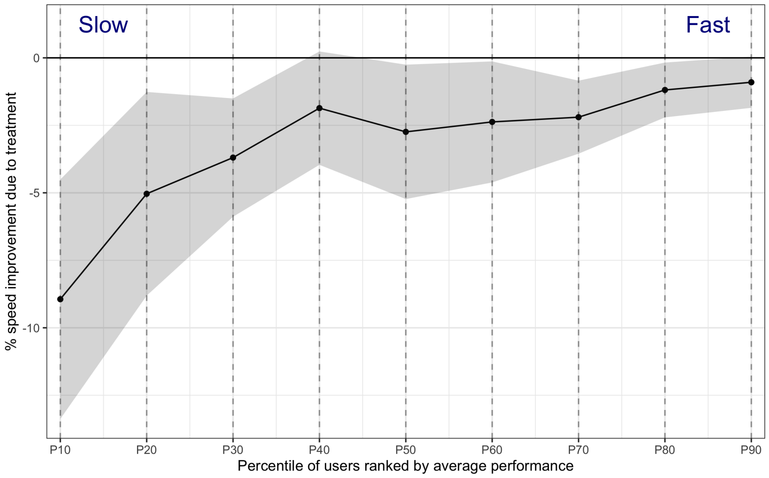

Figure 3: *The experiment impacts fast and slow users differently. Slow users are hurt (in terms of speed) while mid-range and fast users are mostly preserved. Specifically, we use quantile regression to compare performance under the same user-level quantile. Each user-level observation is the 50th-percentile of a single user's observations.*

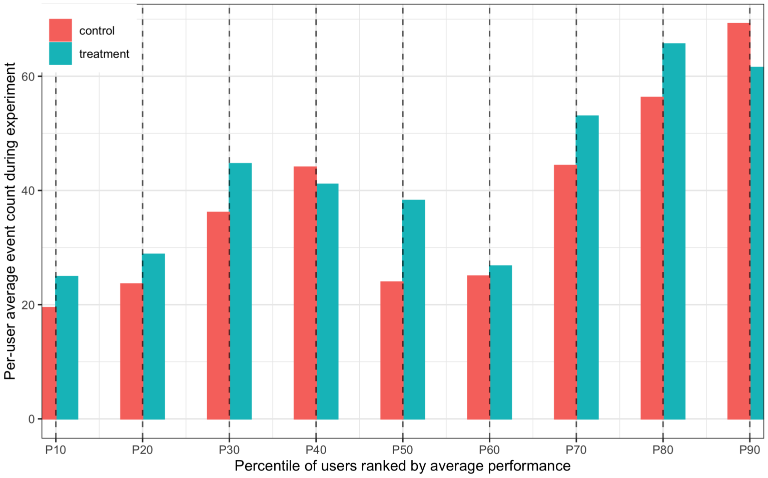

Figure 4: *The majority of events are emitted by users on fast devices who are only minimally impacted. This pulls up event-level results while explaining why event-level better reflects lack of impact on the low-end. For instance, the red bar (P30, 36) indicates the average user in the 30th-percentile of performance in control had 36 performance events during the experiment duration.*

5 Conclusion and future work

In this work we find that mobile performance results in A/B experiments can differ widely depending on whether one measures at the event-level or the user-level. We demonstrate differences between both measurement approaches on 152 randomly selected A/B experiments at Snapchat. We find that 10.06% of A/B test results from comparing these two methods show differences in effect direction, significance or both. We present two experiments where event-level performance metrics only capture the treatment effect on heavy users, which is different from the average treatment effect of all users captured by the user-level performance metrics. This disparity originates from the fact that mobile devices on the market have high heterogeneity and thus users with higher-end devices may have very different experience than users with lower-end devices regarding any single performance change of a mobile app. Thus we recommend tech companies analyze mobile performance on both the event-level and user-level, and in general think carefully about potential heterogeneity that can distort performance measurement in online experiments.

References

[1] An, Daniel. "Find Out How You Stack Up to New Industry Benchmarks for Mobile Page Speed." *Google*, Google, Feb. 2017, www.thinkwithgoogle.com/marketing-resources/data-measurement/mobile-page-speed-new-industry-benchmarks/.

[2] Beales, Joel, and Jeffrey Dunn. "MobileLab: Prevent Mobile Performance Regressions." *Facebook Engineering*, 29 Oct. 2018, engineering.fb.com/android/mobilelab/.

[3] Bui, Duc Hoang, et al. "Rethinking energy-performance trade-off in mobile web page loading." Proceedings of the 21st Annual International Conference on Mobile Computing and Networking. ACM, 2015.

[4] Dean, Jeffrey, and Luiz André Barroso. "The tail at scale." *Communications of the ACM* 56.2 (2013): 74-80.

[5] Diorio, Jon. "Introducing the Mobile Speed Scorecard and Impact Calculator." *Google*, Google, 26 Feb. 2018, www.blog.google/products/ads/speed-scorecard-impact-calculator/.

[6] Eaton, Kit. "How One Second Could Cost Amazon $1.6 Billion In Sales." *Fast Company*, Fast Company, 30 July 2012, www.fastcompany.com/1825005/how-one-second-could-cost-amazon-16-billion-sales.

[7] Fabijan, Aleksander, et al. "Diagnosing Sample Ratio Mismatch in Online Controlled Experiments: A Taxonomy and Rules of Thumb for Practitioners." Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery Data Mining. ACM, 2019.

[8] Grønli, Tor-Morten, et al. "Mobile application platform heterogeneity: Android vs Windows Phone vs iOS vs Firefox OS." 2014 IEEE 28th International Conference on Advanced Information Networking and Applications. IEEE, 2014.

[9] Karumuri, Suman. "Applications of (Pin)Trace Data." *Medium*, Pinterest Engineering Blog, 14 June 2017, medium.com/pinterest-engineering/applications-of-pin-trace-data-3b9e6dc2744b.

[10] Kedar, Shantanu. "New State of Online Retail Performance Report Available." Akamai, 16 May 2018, 7:52 AM, blogs.akamai.com/2018/05/new-state-of-online-retail-performance-report-available.html.

[11] Koenker, Roger, et al. "Package 'quantreg'." (2019).

[12] Kohavi, Ron, et al. "Controlled experiments on the web: survey and practical guide." Data mining and knowledge discovery 18.1 (2009): 140-181.

[13] Loughran, Colin, et al. "Think Fast: The 2019 Page Speed Report Stats & Trends for Marketers." Unbounce, 2019, unbounce.com/page-speed-report/.

[14] Liu, Min, et al. "Large-Scale Online Experimentation with Quantile Metrics." arXiv preprint arXiv:1903.08762 (2019).

[15] Lux, Matthias. "Analyzing Experiment Outcomes: Beyond Average Treatment Effects." *Uber Engineering Blog*, 7 Nov. 2018, eng.uber.com/analyzing-experiment-outcomes/.

[16] Schmohl, Robert, and Uwe Baumgarten. "Heterogeneity in mobile computing environmens." International Conference on Wireless Networks in Las Vegas, USA. 2008.

[17] Smith, Dave. "How to Metric." *Medium*, Medium, 10 Sept. 2018, medium.com/@djsmith42/how-to-metric-edafaf959fc7.

[18] Stephens, David A. *Asymptotic Distribution of Sample Quantiles*. McGill University, 2006, www.math.mcgill.ca/ dstephens/OldCourses/556-2006/Math556-Median.pdf.

[19] Tran, Cuong. "Who Moved My 99th Percentile Latency?" *LinkedIn Engineering*, 8 Apr. 2015, engineering.linkedin.com/performance/who-moved-my-99th-percentile-latency.

[20] Zhang, Yunqi, et al. "Treadmill: Attributing the source of tail latency through precise load testing and statistical inference." Acm Sigarch Computer Architecture News 44.3 (2016): 456-468.