Portfolio X - User Engagement

Introduction

The app, *WordBook*, available on the App Store and Google Play Store, helps users learn new vocabulary through gamification. To optimise for Power-users[1], the developers aim to model user behaviour and analyse the distribution of words covered based on daily participation, time spent, and reading speeds. By seeing consumption patterns of Power-users (80th percentile), I determine the optimal number of high-quality words to display per user i.e. optimising individual engagement with overall word quality while maintaining sufficient vocabulary for future updates (since high-quality words come from a limited pool).

Methodology

To model user engagement, I assume 85% of users open the app daily, which I represent using a Bernoulli distribution (They either use the app, or they don't). For those who do engage, the time spent is modelled with a Gamma distribution. On average, users spend 2 minutes per session, and with an estimated 8 seconds per word. Gamma distribution is appropriate here because it can model positive skewness which is an observed tendency of the Power-users who engage extensively, thus a right-skewed distribution.

1. **Participation:** Each user has a daily probability $P = 0.85$ of using the app, modelled as a Bernoulli random variable:

$$\text{UseApp} \sim \text{Bernoulli}(P).$$

2. **Time Spent:** For active users, the time spent T (in minutes) follows a Gamma distribution with shape parameter k and scale parameter θ:

$$T \sim \Gamma(k, \theta),$$

where the parameters are defined based on the desired mean μ_T and standard deviation σ_T:

$$k = \frac{\mu_T^2}{\sigma_T^2}, \quad \theta = \frac{\sigma_T^2}{\mu_T},$$

[1]Power-users obsessively active users, here the top 20% of users

I use $\mu_T = 2$ minutes (mean) and $\sigma_T = 1$ minute (standard deviation).

3. **Reading Speed:** Each user's reading speed S (in seconds per word) is modelled as a truncated normal distribution to make sure all value are positive:

$$S \sim \mathcal{N}^+(\mu_S, \sigma_S^2),$$

where

$$\mu_S = 8\,\text{seconds/word}, \quad \sigma_S = 4\,\text{seconds/word}.$$

4. **Words Covered:** The number of words an active user covers is calculated as:

$$W = \frac{T \times 60}{S},$$

where T is the time spent (in minutes), converted to seconds, and S is the seconds per word (computed above).

5. **Simulations and Error Bars:** To estimate the distribution of W, I conducted 1,000 independent simulations, each with 100,000 users. The error bars represent $[-\sigma, +\sigma]$ confidence intervals (one full standard deviation!) of the bin frequencies across all simulations.

Simulation

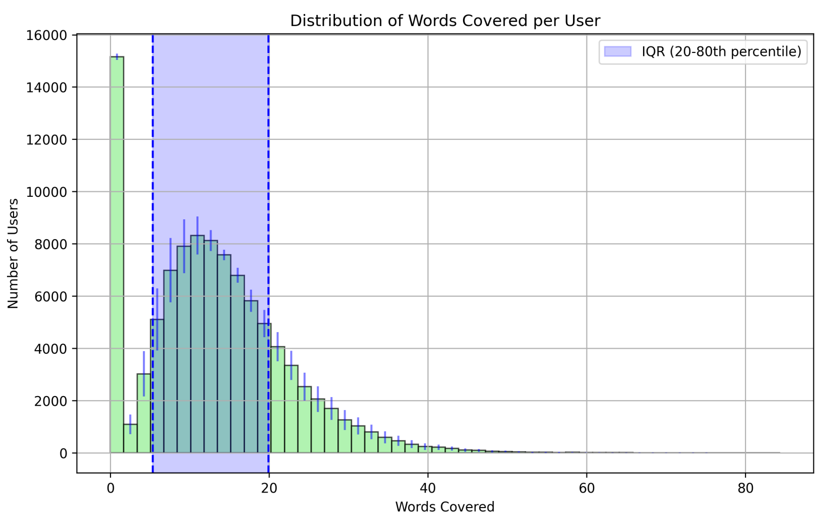

Figure 1: Histogram of words covered per user across 100,000 simulations. Most users cover 5–20 words, with a right-skewed distribution. The blue region represents the interquartile range (20th–80th percentile). Approximately 15% of users do not open the app, as shown by the leftmost bar.

20th percentile: 5.4 words
80th percentile: 19.9 words

What I Learned

- Everything is random. The deeper you go, the more you can model as 'randomness'. Chasing the 'best' model often means settling for one that works. Best to keep it simple. Maybe an infinitely detailed model would perfectly predict every aspect of life and every human's actions, making life deterministic and not stochastic?
- Simple napkin-maths models are quick, easy, and insightful. For our app, knowing we only need to deliver 20 high-quality words per day allows us to save the rest for future use!
- Being able to express creative thinking and apply theories and methods learned in SOR3012 to self-initiated commitments and projects :)