

Title: Effects of Frequency Distribution on Linear Menu Performance
Authors: Wanyu Liu, Gilles Bailly, Andrew Howes
DOI: <https://doi.org/10.1145/3025453.3025707>

Motivation: Before the first lecture I had not given much thought about the possible implications of menu design and was curious about the research. Initially I only intended to skim through the paper, but because of the sheer number of topics that I either strongly disagreed on or greatly questioned during lecture 2, I wanted to dig deeper.

Contribution Type: I would categorize this paper as an empirical study because it discussed and trialed a phenomenon that's been widely accepted in the scientific community (Zipf's law). Additionally, I would classify it as a method contribution since it actually uses zipfian distributions which are often avoided due to their complexity in similar studies.

Scope: This paper trials different distributions for menus and how they impact a human user's ability to locate a desired menu item.

Approach: The trials utilize three different distribution types to test how quickly a correct menu, and subsequent menu item is selected by the human user. A uniform distribution was used to establish a baseline, and then two zipfian distributions with an s-parameter of 1 and 2 respectively as comparisons. The base assumption was that menus that follow a zipfian distribution are more efficient to use due to frequently used items occurring earlier.

Evidence: 24 participants collectively performed around 16K trials which is a reasonable number, but based on how these people were selected, many of them were likely quite similar which could add some form of hidden bias to the results. With that said, even at this scale the study was able to derive that the location of menu is indeed faster to find with a zipfian distribution compared to a uniform distribution, but unintuitively, the reverse applies to the finding menu items.

Generalizability & Alternative Applications: With how universal menus are in software and how universal software itself is, I'm not sure how this could be "broadened" any further.

Learning outcome & Discussion: While I certainly learned some new things and gained perspective on the matter, I'm still not convinced of the methodology. The author's use the time it takes for a user to locate a menu item to assess a distribution's performance, which I am not sure makes sense or is particularly useful for any real world application. Is saving a fraction of a second truly worth the potentially infinite amount of frustration that a "living" dynamic menu would cause? This type of design makes me think that the author's might be "reverse anthropomorphizing" humans by assuming that we work the same as machines. I claim this because the idea sounds perfect for an algorithm but I wouldn't wish it on my worst enemy. Regardless, the contradictory findings were quite eye opening how one type of methodology can work well for one type of menu, but be completely detrimental to another, even when both menus are adjacent.