# A Phrase Dataset with Difficulty Ratings Under Simulated Touchscreen Input

Dylan Gaines
Michigan Technological University
Houghton, MI, USA
dcgaines@mtu.edu

Keith Vertanen
Michigan Technological University
Houghton, MI, USA
vertanen@mtu.edu

## ABSTRACT

We extract phrases from the web forum Reddit for use in text entry studies. We simulate the input of these phrases on a touchscreen keyboard with auto-correct and word completions while using different input noise levels and language model sizes. We rank the difficulty of each phrase from 1–10 based on the character error rate of the simulation. We release the final phrases and metadata to allow researchers to select phrases according to the needs of their study. We conjecture that more difficult phrases will be useful for testing an interface's features designed to help users detect, avoid, or correct recognition errors.

## CCS CONCEPTS

• **Human-centered computing** → **Text input**; **User studies**.

**ACM Reference Format:**
Dylan Gaines and Keith Vertanen. 2022. A Phrase Dataset with Difficulty Ratings Under Simulated Touchscreen Input. In *TEXT2030: MobileHCI'22 Workshop on Shaping Text Entry Research in 2030, October 1, 2022, Vancouver, Canada.* ACM, New York, NY, USA, 3 pages.

## 1 INTRODUCTION

One of the most effective ways to test a novel text entry interface is to have users enter text using it. Many user studies have users copy sentences or phrases that have been previously collected by a researcher, such as the Enron mobile dataset [8] or the MacKenzie phrase set [4]. However, as mobile devices and language evolve, text that was once difficult to enter on a mobile device can become trivial. This can make it difficult for researchers to test the effectiveness of the error prevention or correction features of an interface. One option for solving this problem is to use composition-style tasks, in which users invent their own sentences to type [9] or describe an image [1] instead of copying existing phrases. Gaines et al. [2] showed that users are able to modulate the difficulty of their compositions when asked, and provided a variety of methods for obtaining the users' intended text. However, the difficulty and style of compositions will vary between users and asking them to invent their own phrases can add to the cognitive load of the task. Paek and Hsu [5] varied the difficulty of their phrases based on the perplexity of the phrases, but this method of evaluating difficulty is quite removed from the target task of text entry.


*TEXT2030, October 1, 2022, Vancouver, Canada*


To counteract these issues, we build a phrase set of text composed by users of the online forum Reddit[1]. We simulate noisy input of this text on a typical touchscreen keyboard that offers auto-correct and word completions. We then assign a difficulty rating to each prompt based on the average final error rate experienced by our simulated user. This rating system will allow researchers to select a set of phrases appropriate to the goals of their particular study.

## 2 DATASET CONSTRUCTION

We sourced sentences from a dump[2] of comments from Reddit recorded between December 2005 and August 2019. We dropped any comment that was later marked as deleted. For each comment, we used a set of heuristics to extract likely sentences based on case and punctuation. We dropped any repeated identical sentences in a given comment. We dropped sentences containing characters besides A–Z, apostrophe, comma, period, exclamation mark, and question mark. We also dropped sentences with an out-of-vocabulary (OOV) rate higher than 20% with respect to a large word list of 735 K words obtained from human-edited English dictionaries. We randomized the final set of sentences and selected the first 500 phrases of each length between two and ten words, inclusive. We also selected the first 500 phrases that had a length between eleven and fifteen words, inclusive, bringing the total dataset size to 5000 phrases. We converted phrases to lowercase and removed punctuation except for apostrophes.

We ran this list of phrases through a touchscreen input simulation. We used a keyboard definition based on the iPhone keyboard with the letters A–Z arranged in a Qwerty layout. We added an apostrophe key to the right of the letter M. At each step, our simulation either entered the next character in the phrase or selected a word prediction slot if the intended word occupied one of them. Each entered character was simulated as a tap at the center of the corresponding key, perturbed by some Gaussian noise with a standard deviation equal to a key's size in the x- and y-dimension multiplied by a factor. We used Low, Medium, and High noise levels using factors of 0.25, 0.50, and 0.75 respectively.

Since many commercial soft keyboards include three word suggestion slots, we chose to do the same within our simulation. We selected the configuration of three slots shown to have the best keystroke savings in Vertanen et al. [7] while also including a Literal slot. As described in Vertanen et al. [7], the three slots used were:

- **Literal slot** — The letters nearest to each tap. Vertanen et al. [7] found that this slot helped users enter out-of-vocabulary (OOV) words.

[1] https://www.reddit.com/
[2] https://files.pushshift.io/reddit/comments/

- **Prefix slot** — A word completion based on the currently noisy prefix input of a word. A user's taps thus far are treated probabilistically during the search for likely word completions.
- **Likely slot** — Whatever hypothesis has the highest probability regardless of whether it is a prefix completion or recognition alternatives (similar to prefix completions but the decoder assumes the taps represent an entire word as opposed to a prefix).

All slots were updated by a probabilistic decoder based on VelociTap [10] after each action taken by the simulation. The decoder's search parameters and penalties were configured as described in Vertanen [6]. Since this work found only marginal gains by using a word language model, we chose not to utilize one. We used the Small (26 MB), Medium (125 MB), and Large (607 MB) character language models from Vertanen [6], as well as the 100K vocabulary. Each phrase was run through the simulation 100 times on each combination of noise level and character language model size. The average character error rate (CER) was computed for each phrase as the number of insertions, deletions, and substitutions required to transform the entered text to the reference text, divided by the length of the reference text and multiplied by 100%. We assigned each phrase a difficulty rating from 1 to 10 based on the average CER of each phrase over its 900 simulation runs (3 noise levels × 3 language models × 100 noisy samples) using the CER cutoffs shown in Table 1.

## 3 DATASET DESCRIPTION

We include in our released *Reddit difficulty-rated phrase set*[3] our full set of phrases, as well as a recommended set. The recommended set consists of the first 100 phrases of each difficulty that both authors agreed contained no obvious typos or offensive text, and that were not unduly confusing when read as a sentence in isolation. Each set is presented in a tab-separated format with the following columns:

- phraseID — A unique ID for the phrase.
- phrase — The lowercase phrase with punctuation removed (except for apostrophes).
- difficulty — The overall difficulty rating between 1 and 10 (described in the previous section).
- numWords — The number of words in the phrase.
- numChars — The number of characters in the phrase including spaces.
- longestWord — The number of characters in the longest word.
- <noise><Size>CER — A set of columns (e.g. lowSmallCER) representing the average CER across the 100 trials with the specified noise level and language model size.
- <noise>NoiseAvgCER — A set of columns (e.g. lowNoiseAvgCER) representing the average CER across all trials and language model sizes with the specified noise level.
- overallAvgCER — The overall CER across all combinations of noise and model size.
- oovCount — The number of words not in the 100K vocabulary used.
- maxRank — The maximum rank of a word in the phrase in a unigram frequency list.
- profane — 1 if the phrase includes potentially profane words, 0 otherwise. This is based on a list of 1,747 words and may not have caught all profanity.
- charPerplexity<Size> — A set of columns (e.g. charPerplexitySmall) representing the character perplexity under the specified model.
- wordPerplexityLarge — The word perplexity under a large word language model.
- originalPhrase — The phrase with its original capitalization and punctuation.

[3] https://keithv.com/data/rated.zip

## 4 CONTRIBUTION

The text entered on mobile devices has changed drastically over the last decade and will likely continue to do so over the next decade. As this text changes, our text entry research studies need a way to change with it. By developing and releasing this phrase set as a public tool with which to evaluate text entry systems, we hope to introduce more modern text into transcription tasks. The methodology that we use here can be repeated on text obtained from future forum messages as needed to ensure that our transcription prompts stay up-to-date. It is also likely that as our language models and decoders evolve, we will see a shift in what is difficult for a decoder to decipher accurately, creating the need for difficulty ratings to be adjusted.

## 5 DISCUSSION AND CONCLUSIONS

While our difficulty ratings can be useful for researchers to tune the phrase set to their research questions, they are based solely on simulated input. Further work could be done to validate or rescore the difficulty ratings based on corrective behaviors performed by users during actual text entry. Our difficulty ratings were also only calculated with respect to touchscreen input on a Qwerty keyboard; other keyboard layouts or entry methods such as speech recognition could produce different results than we found here. Additionally, our simulation used synthetic noise causing possible substitution errors. It did not simulate missing or extra touch events as sometimes occur during touchscreen text input. A higher fidelity simulation might use touch events sampled from actual user data with occasional touch insertions and deletions. Another limitation is that we did not test the memorability of any of our phrases using a method similar to Leiva and Sanchis-Trilles [3].

That being said, we feel that this phrase set is quite versatile. In an experiment on a smartwatch device, researchers could filter phrases based on the CER from our high noise and small model simulations. In an experiment on a tablet, researchers could use the low or medium noise simulations to find phrases that are difficult even with less noise. Moving into a future of text input on a variety of new devices (e.g. augmented and virtual reality headsets), it is essential that we are able to tune our phrase set to suit the needs of our studies.

## ACKNOWLEDGMENTS

This material is based upon work supported by an NSF Graduate Research Fellowship (2034833).

| Difficulty | CER | Number of Phrases | Avg Char Perplexity | Example Phrase |
|---|---|---|---|---|
| 1 | 7.00% | 454 | 2.266 | yeah that's a good idea |
| 2 | 8.50% | 600 | 2.509 | i personally have to classify my favorites |
| 3 | 9.50% | 499 | 2.724 | i've taken a screenwriting class before |
| 4 | 10.25% | 413 | 2.898 | this is my ultimate bias |
| 5 | 11.25% | 533 | 3.058 | yeah i would avoid arguing too |
| 6 | 12.25% | 490 | 3.276 | huh they look about the same to me |
| 7 | 13.50% | 455 | 3.501 | its different when howard does it |
| 8 | 15.50% | 568 | 3.889 | i'll take my xmas present early |
| 9 | 18.50% | 470 | 4.474 | and comes with gps |
| 10 | > 18.50% | 518 | 6.990 | steelix is the best legendary |

**Table 1: Summary statistics about each difficulty level, including the character error rate at or below which a prompt is classified as a particular level. Also displayed are the number of phrases, the average character perplexity (under the LARGE model), and a sample phrase for each level.**

## 6 AUTHOR BIOGRAPHIES

**Dylan Gaines**. Dylan Gaines is a Ph.D. candidate and National Science Foundation Graduate Research Fellow at Michigan Technological University. The main focus of his research is on systems that utilize non-visual methods of text entry.

**Keith Vertanen**. Keith Vertanen is an Associate Professor at Michigan Technological University. He specializes in designing intelligent interactive systems that leverage uncertain input technologies. This includes input via speech, on touchscreens, in mid-air, and via eye-gaze. A particular focus of his research is on systems that enhance the capabilities of users with permanent or situationally-induced disabilities.

## REFERENCES


[1] Mark D. Dunlop, Emma Nicol, Andreas Komninos, Prima Dona, and Naveen Durga. 2016. Measuring Inviscid Text Entry Using Image Description Tasks.

[2] Dylan Gaines, Per Ola Kristensson, and Keith Vertanen. 2021. Enhancing the Composition Task in Text Entry Studies: Eliciting Difficult Text and Improving Error Rate Calculation. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 725, 8 pages. https://doi.org/10.1145/3411764.3445199

[3] Luis A. Leiva and Germán Sanchis-Trilles. 2014. Representatively Memorable: Sampling the Right Phrase Set to Get the Text Entry Experiment Right. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 1709–1712. https://doi.org/10.1145/2556288.2557024

[4] I. Scott MacKenzie and R. William Soukoreff. 2003. Phrase Sets for Evaluating Text Entry Techniques. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems* (Ft. Lauderdale, Florida, USA) *(CHI EA '03)*. Association for Computing Machinery, New York, NY, USA, 754–755. https://doi.org/10.1145/765891.765971

[5] Tim Paek and Bo-June (Paul) Hsu. 2011. Sampling Representative Phrase Sets for Text Entry Experiments: A Procedure and Public Resource. In *CHI 2011* (chi 2011 ed.). ACM. https://www.microsoft.com/en-us/research/publication/sampling-representative-phrase-sets-for-text-entry-experiments-a-procedure-and-public-resource-2/

[6] Keith Vertanen. 2021. *Probabilistic Text Entry-Case Study 3* (1 ed.). Association for Computing Machinery, New York, NY, USA, 277–320. https://doi.org/10.1145/3447404.3447420

[7] Keith Vertanen, Dylan Gaines, Crystal Fletcher, Alex M. Stanage, Robbie Watling, and Per Ola Kristensson. 2019. VelociWatch: Designing and Evaluating a Virtual Keyboard for the Input of Challenging Text. In *CHI '19: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland).

[8] Keith Vertanen and Per Ola Kristensson. 2011. A Versatile Dataset for Text Entry Evaluations Based on Genuine Mobile Emails. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services* (Stockholm, Sweden) *(MobileHCI '11)*. Association for Computing Machinery, New York, NY, USA, 295–298. https://doi.org/10.1145/2037373.2037418

[9] Keith Vertanen and Per Ola Kristensson. 2014. Complementing Text Entry Evaluations with a Composition Task. *ACM Transactions on Computer-Human Interaction* 21, 2, Article 8 (2014), 8:1–8:33 pages.

[10] Keith Vertanen, Haythem Memmi, Justin Emge, Shyam Reyal, and Per Ola Kristensson. 2015. VelociTap: investigating fast mobile text entry using sentence-based decoding of touchscreen keyboard input. In *CHI '15: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Seoul, Korea). 659–668.