Rethinking Tokenization: Crafting Better Tokenizers for Large Language Models

Jinbiao Yang

December/2023

摘要

Tokenization significantly influences large language models(LLMs)' performance. This paper traces the evolution of tokenizers from word-level to subword-level, analyzing how they balance tokens and types to enhance model adaptability while controlling complexity. Despite subword tokenizers like Byte Pair Encoding (BPE) overcoming many word tokenizer limitations, they encounter difficulties in handling non-Latin languages and depend heavily on extensive training data and computational resources to grasp the nuances of multiword expressions (MWEs). The study introduces the "Principle of Least Effort" from cognitive science, that humans naturally seek to reduce cognitive effort, and applies the principle to language learning and usage. Based on this principle, the paper proposes that the Less-is-Better (LiB) model could a new approach for LLM tokenizer. The LiB model can autonomously learn an integrated vocabulary consisting of subwords, words, and MWEs, which effectively reduces both the numbers of tokens and types. Comparative evaluations show that the LiB tokenizer outperforms existing word and BPE tokenizers, presenting an innovative method for tokenizer development grounded in cognitive science. This strategy, emulating human language processing's versatility, hints at the possibility of future tokenizers being more efficient and adaptable across various languages and providing improved semantic accuracy.

1 Introduction

When confronted with vast or intricate information, our brains typically simplify it into smaller, more digestible segments, thereby helping us better understand and remember. Language, exemplifying such complexity, often requires segmenting itself into *tokens* through a process known as *tokenization*.

In the field of natural language processing (NLP), the choice of tokenizer has a crucial impact on the performance of language models. Especially in large language models (LLMs), how a tokenizer segments corpora determines the fundamental way the model processes language. This article investigate the roles of tokens (the actual number of lexical units in a corpus) and types (the number of different lexical units of vocabulary) in tokenizer design, and attempts to find an ideal solution that optimizes the number of tokens while controlling the number of types. In the following sections, we will explore the advantages and limitations of subword tokenizers, analyze the treatment of Multiword Expressions

(MWE) in current large language models. Then we will explain the "Principle of Least Effort" in human language acquisition, and introduce a new type of tokenizer model - the *Less-is-Better* model, based on the Principle of Least Effort.

1.1 From Word-level Tokenizers to Subword-level Tokenizers

NLP applications initially relied on word-level tokenizers, which divided text into words using spaces and punctuation. For example, the historical development of semantic representations started with the Bag-of-Words model, progressed to Word2Vec by Mikolov et al. (2013), and to GloVe (Global Vectors for Word Representation) by Pennington et al. (2014). They all aimed at training semantic representations at the level of words. Word-level tokenizers are relatively effective in processing European languages, where spaces provide clear word boundaries. However, this method is limited in languages like Chinese, which do not have clear word boundaries. Moreover, the flexible morphological inflections in language, the constant emergence of new words, and the prevalence of spelling errors in corpora make it difficult for word-level vocabularies to generalize in practical applications.

Subword technology can be traced back to the 1990s (Gage, 1994). Initially, these techniques were mainly used to compress data. With the emergence of LLMs, the demand for tokenizers increased. These complex models require an understanding and generation of extremely rich and diverse language content, and traditional word-level tokenizers struggle with complex vocabularies, morphological inflections, and the continuous influx of new vocabularies. At this point, subword-level tokenizers became the new mainstream due to their flexibility and generalization capabilities.

表 1: Popular tokenization methods that contributed to the evolution of language models in recent years.

Tokenizer	Representative Papers	Year	Used in Notable Models
BPE	"Neural Machine Translation of Rare Words with Subword Units" (Sennrich et al., 2016)	2016	GPT-2&3&4
SentencePiece	"SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing" (Kudo & Richardson, 2018)	2018	ALBERT (Lan et al., 2020), T5 (Raffel et al., 2019), XLNet (Z. Yang et al., 2020)

表 2: Examples of text segmentation using BPE. The tokenizer used is provided officially by OpenAI for GPT-3.5 and GPT-4 (https://platform.openai.com/tokenizer).

Categories	Examples
English text	Generative Pre-trained Transformer 4 (GPT-4) is a multimodal large language model created by
English subwords	Gener|ative| Pre|-trained| Transformer| |4| (|G|PT|-|4|)| is| a| multim|odal| large| language| mod
Chinese text	您可以使用下面的工具了解语言模型如何对一段文本进行标记化，以及这段文本中的标记总数。
Chinese subwords	您 | 可以 | 使用 | 下 | 面 | 的 | 工 | 具 | 了 | 解 | 语 | 言 | 模 | 型 | 如 | 何 | 对 | 一 | 段 | 文 | 本 | 进

Tokenizer	Representative Papers	Year	Used in Notable Models
Unigram	"Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates" (Kudo, 2018)	2018	T5 (Raffel et al., 2019)
WordPiece	"Japanese and Korean Voice Search" (Schuster & Nakajima, 2012);"BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding" (Devlin et al., 2019)	2019	BERT (Devlin et al., 2019), ERNIE (Sun et al., 2019)

1.2 Balancing Tokens and Types by Subwords

In the transition from word-level to subword-level, a core consideration is how to balance the number of tokens and types. Word-level tokenizers, although producing fewer types, are unable to deal with Out-Of-Vocabulary (OOV) units. In contrast, subword-level tokenizers significantly reduce the occurrences of OOV, enhancing the model's adaptability to new vocabularies and complex language phenomena.

For example, BPE and WordPiece, in creating their vocabularies, effectively handle rare vocabularies by gradually merging frequently occurring character pairs or combinations, while keeping the number of tokens within a reasonable range. SentencePiece and Unigram models further improve adaptability to different languages, especially in languages that do not use spaces to separate words (like Chinese).

As shown in the comparison in Table 1, the number of types in BPE is 4.7% of words, while the number of tokens is roughly equal (111%); the number of types in characters is 0.2% of words, but the number

表 3: An example of the number of tokens/types with different tokenizers on a German corpus (See Table 1 in @Sennrich2016-fv).

Segmentation	Tokens	Types
Words	100 million	1,750,000
BPE	111 million	82,000
Characters	550 million	3,000

of tokens is 550%. This shows that the subword approach is high-yield (significantly reduced number of types) and low-cost (slightly higher number of tokens).

For languages with rich morphology (like the German corpus shown in Table 3), the significant reduction in the number of types with subword-level tokenization is mainly because it can break down words into frequent subunits, capturing morphological variations without needing separate entries for each word form. Although Chinese does not have a lot of morphological variations, the presence of numerous compound words (composed of two or more morphemes, like "关闭" [shut down], "直升机" [helicopter]) means that subword-level tokenization can also reduce the number of types.

The reduction in types lowers the computational complexity and memory requirements of LLMs, having a direct positive impact on the models' performance. Moreover, subword-level tokenization has a stronger generalization capability for OOV contents or spelling errors in the corpus, as shorter tokens have a higher probability to cover more of the corpus. Thus, LLMs generally adopt the subword approachs for alphabetic languages (Table 1).

This shift from word-level to subword-level not only marks the progress of tokenizer technology in the NLP field but also reflects a deeper understanding of language diversity and complexity. However, as seen in the examples in Table 2, unlike languages using the Latin alphabet, Chinese BPE subwords are mostly single characters. An analysis[1] shows that a sentence in Chinese may require 1.7x more tokens than a similar sentence in English, and Burmese or Amharic may require 10x more tokens. This indicates that for LLMs in various languages, especially non-Latin ones (like Chinese), BPE subwords still have shortcomings. Furthermore, although subword tokenizers have made significant progress in handling OOV issues, they still have limitations in capturing the nuanced semantics and idiom implications of language. This leads to the need for direct handling of multiword expressions as a complement and refinement of current tokenizer technology.

1.3 Current Marginalization of Multiword Expressions (MWEs) in Language Models

Multiword Expressions, despite playing a crucial role in everyday language, are often overlooked in the development of large language models (LLMs). So far, only AI21 Studio's Jurassic-X models (Lieber

[1] https://www.artfish.ai/p/all-languages-are-not-created-tokenized

et al., 2021) has introduced multi-word tokens, including expressions, phrases, and named entities, into their vocabularies. This marginalization may be primarily due to several reasons:

1. **Performance and complexity considerations**: Introducing MWEs as independent tokens will obviously increase the number of types. The vocabulary of the aforementioned Jurassic model includes about 250,000 types, much larger than most existing vocabularies (5 times or more)[2]. However, MWEs can be rare or highly specific to certain contexts or domains, so their introduction as independent tokens does not significantly reduce the total number of tokens. This somewhat contradicts the goal of efficient performance pursued by LLMs. Moreover, low-frequency MWEs lead to insufficient representation in the training data, making it difficult for the models to learn and accurately predict their semantics.
2. **The alternative role of big data and computational power**: Current LLMs, like GPT-series and BERT, rely on massive training data and high computational power to learn the real usages of MWEs rather than their literal expressions, even though these expressions are not treated as whole units during training (Tian et al., 2023).

Despite this, the direct recognition and processing of MWEs still have unique values in LLMs: 1. **MWEs can have unique holistic semantics**: Incorporating MWEs with unique holistic semantics, like "kick the bucket" or "摸鱼" [underwork (actual meaning); touching fish (literal meaning)], can enrich the model's language comprehension capabilities. Although this may not significantly reduce the number of tokens, it allows the model to capture the specific semantics of texts containing these MWEs more directly and accurately. 2. **Some MWEs can reduce the number of types**: In some cases, by appropriately selecting MWEs, it might even be possible to reduce the total number of types. For instance, treating common fixed phrases as single units (like "鹦鹉" [parrot], "乒乓" [ping pong]) might reduce the need for their individual parts.

Due to the vast amount and diversity of MWEs, there was a scarcity of MWE lexicons. This scarcity consequently hindering their integration into current development of large language models. However, linguists and psycholinguists have long studied on MWEs , and we can rediscover their value based on human cognition:

- **Combining model performance with human language cognition**: With technological advancements, especially when LLMs reach engineering limits, LLMs can draw more from human language cognition processes.
- **Beyond pure computational power**: Although big data and powerful computing can solve MWE processing to a certain extent, this "brute force" method might not be as efficient and precise as a carefully designed LLM/tokenizer that can directly handle MWEs. Like the attention mechanism (Vaswani et al., 2023) for Deep Learning and Reinforcement Learning from Human Feedback (RHLF) (Ouyang et al., 2022) for LLMs, we can reconsider drawing inspiration from human language cognition processes when reaching engineering limits.

[2]https://www.ai21.com/blog/announcing-ai21-studio-and-jurassic-1

While the tokenization of subwords and MWEs can offer certain advantages for LLMs, exploring deeper principles of language processing is still needed in understanding and optimizing tokenizers. This leads us to a core principle of human language acquisition and use - the "Principle of Least Effort".

2 Optimizing Future Tokenizers

2.1 Principle of Least Effort

Tokenizers, more than mere technical tools, emulate and learn from human language processing methods. Therefore, in the quest to optimize tokenizer design, one cannot overlook the mechanisms of human language processing. As a theory of human language processing, George Kingsley Zipf's Principle of Least Effort, articulated in his 1949 book "Human Behavior and the Principle of Least Effort" (Zipf, 1949) is fundamentally a statistical observation about language and other human behavior systems, stating that people tend to follow the path that minimizes effort.

Applying the Principle of Least Effort in language suggests seeking minimization of cognitive burden in language learning and usage. This principle manifests in the process of humans learning language from simple to complex: initially, humans tend to use and learn shorter tokens, which gradually become larger and more complex as cognitive abilities develop. We can refer these gradually expanding, cognitively basic units as "*cognitive units*" (J. Yang, 2022).

Unlike fixed linguistic categories like subwords, words, or MWEs, *cognitive units* are characterized by their adaptability in size and form. This adaptability is evident in how infants and illiterate individuals acquire language - they acquire and use various forms of cognitive units from their environment, even without a formal understanding of what a "word" is. This observation hints at an unsupervised(innate), or at semi-supervised computational mechanism in the brain, capable of autonomously extracting suitable cognitive units from language input. Taking the Principle of Least Effort as an optimization goal, an unsupervised implementation of the aforementioned computational mechanism can be achieved.

Zipf's original expression of this principle is encapsulated in his book (Zipf, 1949, p. 1) that "[the person] will strive to solve his problems in such a way as to minimize the total work that he must expend in solving both his immediate problems and his probable future problems." The mention of "immediate problems" and "probable future problems" are crucial, as they encompass both short-term and long-term needs to reduce burden, which may be conflicting. Thus, following the Principle of Least Effort means achieving a balance where each cognitive module pursues its own minimal burden. In the previous sections, we evaluated various tokens based on the number of Types and Tokens, and through the lens of the Principle of Least Effort: - Fewer tokens can lessen the cognitive burden of working memory storage and information decoding steps; - Fewer types can reduce the number of types helps alleviate the burden of long-term memory storage and retrieval.

LLM compression theory and the Principle of Least Effort: The Chief Scientist of

OpenAI, Ilya Sutskever, explained in his talk [3] that unsupervised learning and LLMs can be viewed as approximating optimal data compression. This aligns with the Principle of Least Effort. Using Minimum Description Length (MDL) theory (Rissanen, 1978), we can see that fewer types represent a more compressed description of the encoder model, and fewer tokens represent a more compressed description of the encoded data. Compression seeks the minimal total of the encoder model and encoded data. It is worth noting that a crucial difference from this data compression theory is that each brain module seeks its own minimal burden until the global balance is achieved, since the brain's various areas work in coordination and competition, not necessarily managed by a single global controller.

Applying the 'Principle of Least Effort' to tokenizer design means finding a language processing method that minimizes cognitive burden. This principle can help us understand the shift from word-level tokenizers to subword-level tokenizers, and also supports the introduction of MWEs and the design of more effective tokenizer for LLMs.

2.2 LiB Model: An Implementation of "Principle of Least Effort"

In response to the limitations of existing tokenizer technologies in LLMs, we have introduced a new tokenizer design based on the Principle of Least Effort in the last section. The Least Effort itself, being a principle, can be implemented in various ways. In previous studies(J. Yang et al., 2020, 2022), we proposed an implementation focused on reducing the burden of working memory (number of tokens) and long-term memory (number of types), namely the Less-is-Better (LiB) model. This model aims to mimics the flexible mechanism of human cognitive unit learning. It breaks through the barriers in defining various linguistic units through unsupervised methods, and unifies subwords, words, and multi-word expressions (MWEs) into the same vocabulary. In this process, it effectively balances the number of tokens and types to reduce the cognitive burden of using language (Figure 1).

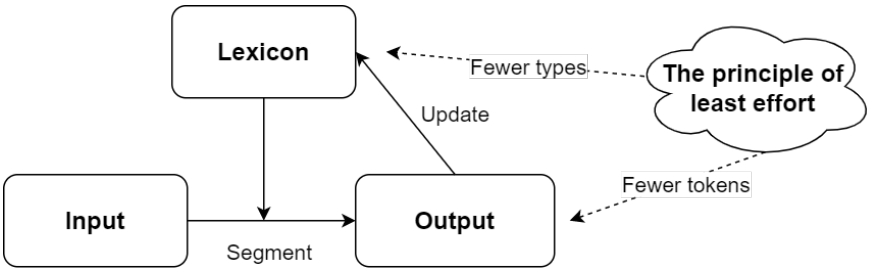

图 1: Information flow in the LiB model.

Model Mechanism: The model consists of a "Memorizer" and a "Forgetter". Initially, the LiB model splits the input corpus into the smallest tokens and then the "Memorizer" continuously merges adjacent tokens in the corpus into new (longer) units and stores them in the vocabulary. By using longer units,

[3] https://www.youtube.com/watch?v=AKMuA_TVz3A

the number of unit tokens in the text decreases, while the number of types increases. Conversely, the "Forgetter" removes less useful "junk" units from the vocabulary to reduce the number of unit types. "junk" units may be those types that increase the number of unit tokens in sentences or are infrequently appearing types. The "Memorizer" and "Forgetter" balance each other, eventually reaching a relatively steady state, where the vocabulary contains units close to the goal of cognitive burden minimization.

Vocabulary Structure: The LiB model's unsupervised method ignores the definition barriers between various traditional linguistic units, so its vocabulary also breaks through the usual limitations of subwords, words, and multi-word expressions. For instance, the units autonomously learned by the model can include various English units like "ly," "you," and "you can", , as well as Chinese units like "的" (English translation: "'s"), "孩子" (English translation: "kid"), and "新华社" (English translation: "Xinhua News Agency") (J. Yang et al., 2020). This fusion reflects the LiB model's flexibility in learning cognitive units of different sizes and linguistic levels.

Practical Application: For corpora in different languages, the LiB model flexibly learns their lexicons through the unsupervised method based on the Principle of Least Effort, thereby adapting to the complexity and diversity of different language inputs, while balancing cognitive loads. In a previous study, it was observed that compared to word-level tokenizers and BPE tokenizers, the units generated by LiB manage to reduce both the number of tokens and types simultaneously (see J. Yang et al. (2020),Table 4). Although LiB is only a cognitive model and has not been optimized for language models, evaluations on simple language models show that LiB-generated units perform better in Bits-per-character scores (see Table 5 in J. Yang et al. (2020)). This suggests the value of the Principle of Least Effort in this era of LARGE language models. We may use the LiB model or other variants that also follow the Principle of Least Effort as tokenizers for large language models to enhance their performance.

This cognitive science-based approach provides a new perspective and direction for the future development of language models, especially when dealing with corpora in various languages (like Chinese, which lacks clear word boundaries).

3 Summary

This article explores the current choice and future optimization of tokenizers for large language models (LLMs), especially in handling complex languages like Chinese. Overall, subword tokenization, as a balancing technique, significantly reduces the number of types while only slightly increasing the number of tokens compared to word tokenization, effectively addressing Out-Of-Vocabulary (OOV) issues and enhancing the model's generalization capabilities. However, this method has limitations in controling the number of tokens in some non-Latin languages (like Chinese), and also in capturing the nuanced semantics and idiom implications of language.

The absence of MWEs in most LLMs reflects a blind spot in the current NLP field. Although MWEs significantly increase the number of types, and current models can learn the meanings of MWEs on

subwords tokenization by massive data/computational power, direct recognition and processing of MWEs can still help language models improve the accuracy in language understanding. In future development of tokenizers, how to effectively select MWEs and balance the number of tokens and types could be a key area for tokenizer advancement.

To address the issues of current tokenizer technologies, we discussed the "Principle of Least Effort" in human language acquisition, which not only reveals efficiency and simplicity in human language processing but also inspires the design of more efficient tokenizers. Based on this principle, we proposed the LiB model, a model that attempts to optimize its vocabulary through learning and forgetting mechanisms, achieving a more effective balance of tokens and types. It aims to simulate human language processing mechanisms, reducing cognitive burden, and obtaining new types of linguistic cognitive units that integrate subwords, words, and MWEs, thereby enhancing the efficiency and accuracy of language processing. The LiB model is not only a reflection on human language processing mechanisms but also provides new ideas for designing more effective tokenizers for LLMs. This cognitive science-based approach provides new perspectives and directions for the future development of tokenizers and language models. Incorporating insights from cognitive science with the design of large language models may enhance their synergistic evolution.

References

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. https://doi.org/10.18653/v1/N19-1423

Gage, P. (1994). A new algorithm for data compression. *The C Users Journal Archive*. https://www.semanticscholar.org/paper/A-new-algorithm-for-data-compression-Gage/1aa9c0045f1fe8c79cce03c7c14ef4b4643a21

Kudo, T. (2018). Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates. In I. Gurevych & Y. Miyao (Eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 66–75). Association for Computational Linguistics. https://doi.org/10.18653/v1/P18-1007

Kudo, T., & Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 66–71. https://doi.org/10.18653/v1/D18-2012

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2020, February 8). *ALBERT: A Lite BERT for Self-supervised Learning of Language Representations*. https://doi.org/10.48550/arXiv.1909.11942

Lieber, O., Sharir, O., Lenz, B., & Shoham, Y. (2021). Jurassic-1: Technical details and evaluation. *White Paper. AI21 Labs*, *1*.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations*

in vector space. http://arxiv.org/abs/1301.3781

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., & Lowe, R. (2022, March 4). *Training language models to follow instructions with human feedback*. https://doi.org/10.48550/arXiv.2203.02155

Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global vectors for word representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. https://doi.org/10.3115/v1/D14-1162

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2019). *Exploring the limits of transfer learning with a unified Text-to-Text transformer*. http://arxiv.org/abs/1910.10683

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, *14*(5), 465–471. https://doi.org/10.1016/0005-1098(78)90005-5

Schuster, M., & Nakajima, K. (2012). Japanese and Korean voice search. *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5149–5152. https://doi.org/10.1109/ICASSP.2012.6289079

Sennrich, R., Haddow, B., & Birch, A. (2016). Neural machine translation of rare words with subword units. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1715–1725. https://doi.org/10.18653/v1/P16-1162

Sun, Y., Wang, S., Li, Y., Feng, S., Chen, X., Zhang, H., Tian, X., Zhu, D., Tian, H., & Wu, H. (2019, April 19). *ERNIE: Enhanced Representation through Knowledge Integration*. https://doi.org/10.48550/arXiv.1904.09223

Tian, Y., James, I., & Son, H. (2023). How Are Idioms Processed Inside Transformer Language Models? In A. Palmer & J. Camacho-collados (Eds.), *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)* (pp. 174–179). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.starsem-1.16

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2023, August 1). *Attention Is All You Need*. https://doi.org/10.48550/arXiv.1706.03762

Yang, J. (2022). *Discovering the units in language cognition: From empirical evidence to a computational model* [PhD thesis, Radboud University & Max Planck Institute for Psycholinguistics]. https://doi.org/10.13140/RG.2.2.35086.84804

Yang, J., Frank, S. L., & van den Bosch, A. (2020). Less is Better: A cognitively inspired unsupervised model for language segmentation. *Proceedings of the Workshop on the Cognitive Aspects of the Lexicon*, 33–45. https://www.aclweb.org/anthology/2020.cogalex-1.4

Yang, J., van den Bosch, A., & Frank, S. L. (2022). Unsupervised text segmentation predicts eye fixations during reading. *Frontiers in Artificial Intelligence*, *5*. https://doi.org/10.3389/frai.2022.731615

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., & Le, Q. V. (2020, January 2). *XLNet: Generalized Autoregressive Pretraining for Language Understanding*. https://doi.org/10.48550/

arXiv.1906.08237

Zipf, G. K. (1949). *Human behavior and the principle of least effort* (Vol. 573). Addison-Wesley Press. https://psycnet.apa.org/fulltext/1950-00412-000.pdf