Relating phone string length and the number of possible word competitors

Matthew C. Kelley

Introduction

The process in question is the degree to which the n-th phone in a word clarifies the word's identity. That is, once the n-th phone has been recognized, how many words are still in contention to be recognized? Alternatively, for a prefix of length n, how many words share that prefix? Note that prefix is used here not in the most traditional liguistic sense of an affix, but in a more general sense of the first n characters of a string, e.g., [di] is a prefix of [di] 'dee', [did] 'deed', [dil] 'deal', [din] 'dean', etc.

Consider a prefix tree representing all possible words in a lexicon. After a phone is experienced, only the subtree corresponding to hearing that phone is still active. After hearing the first phone, a massive number of words are eliminated from contention because they don't start with that phone. After hearing the second phone, a comparatively smaller number of words are eliminated. This pattern continues, where each subsequent phone removes fewer and fewer words from contention because the number of words that match what has been heard srhinks as each successive phone is experienced.

This asymptotic decreasing behavior is reminiscent of a Zipfian relationship. In fact, the relationship between the number of words that share a prefix w_n and the length of the prefix n may well be Zipfian or hyperbolic in nature. To prove that this relationship is true, we would need to prove that $w_n \propto \frac{1}{n}$, where w_n is the number of words still active after hearing the n-th phone. Note that some number being proportional to $\frac{1}{n}$ is only an algebraic manipulation of how Zipf (1949) describes his namesake relationship.

Developing a complete prefix tree for a specific human language is difficult if not impossible to do, let alone for all human languages or human language generally. However, we can contrive an example that is obviously more complex that human language to get an upper bound on the behavior of w_n in real human language. We can additionally contrive an example that is obviously less complex than human language to get a lower bound on the behavior of w_n. If these bounds are tight enough, we may be able to deduce logically that indeed $w_n \propto \frac{1}{n}$.

One aspect of the notation involved in the following proofs requires some additional explanation for those linguists not familiar with algorithmic or asymptotic analysis. This is the "big-O" notation. It is commonly used in computer science to describe the asymptotic bounds of the space- and/or time-complexity of algorithms as a function of the size of the input to the algorithm. It is useful here as well, however, because we are trying to assert asymptotic bounds on w_n, namely, that it is bound both beneath and above by $\frac{1}{n}$, notwithstanding a constant factor. The notations used are "big-O" ($\mathcal{O}$), which describes the upper bounds of a function; "big-Omega" (Ω), which describes the lower bounds of a function; and "big-Theta" (Θ), which describes both the upper and lower bounds of a function.

Formally, if a function $f(n)$ is $\mathcal{O}(g(n))$ (read "big-O of $g(n)$"), there exists some constant $c > 0$ such that $f(n) \leq c \cdot g(n)$. If a function $f(n)$ is $\Omega(g(n))$ (read "big-Omega of $g(n)$)"), there exists some constant $c > 0$ such that $f(n) \geq c \cdot g(n)$. Finally, if a function $f(n)$ is $\Theta(g(n))$ (read "big-Theta of $g(n)$"), there exist two constants $c_1 > 0$ and $c_2 > 0$ such that $c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n)$. Based on these definitions, it follows that if $f(n)$ is both $\mathcal{O}(g(n))$ and $\Omega(g(n))$ that $f(n)$ is also $\Theta(g(n))$. Note that the definition of $f(n)$ being $\Theta(g(n))$ also implies that $f(n)$ is proportional to $g(n)$ ($f(n) \propto g(n)$), scaled by a constant.

Our goal is now clear: we must prove that some function $f(n)$ which yields w_n, the number of words still in contention after hearing the the n-th phone, is $\Theta(\frac{1}{n})$. To prove that $f(n)$ is $\Theta(\frac{1}{n})$, we will prove that $f(n)$ is both $\mathcal{O}(\frac{1}{n})$ and $\Omega(\frac{1}{n})$. More information on big-O notation, as well as some of the inductive proof techniques used here are provided in Cormen et al. (2009).

Formally, we wish to prove the following theorem:

Theorem 1. *In a human language, the function $f(n)$ that that maps between the number of phones in a prefix n and the number of words that share that prefix w_n is $\Theta(\frac{1}{n})$.*

We can prove Theorem 1 by proving the following lemmata:

Lemma 1. *In a human language, the function $f(n)$ that that maps between the number of phones in a prefix n and the number of words that share that prefix w_n is $\mathcal{O}(\frac{1}{n})$.*

Lemma 2. *In a human language, the function $f(n)$ that that maps between the number of phones in a prefix n and the number of words that share that prefix w_n is $\Omega(\frac{1}{n})$.*

Upper bound

Let's begin with the upper bound, Lemma 1. We must prove that $f(n)$ is $\mathcal{O}(\frac{1}{n})$. To contrive the clearly more complex than human language example for our upper bound, we must lay out some assumptions. Assume that all potential

phone strings are a word in the contrived language, and there are no phonotactic constraints on how to combine phones. Assume there are 500 phones in this language. That this is not observed in the world's language's is irrelevant since we are only seeking to find an upper bound. Also assume that that a word can have up to 500 phones in it. If we were to create a prefix tree that is 500 levels deep, we would see that each node branches into 500 subtrees. There are 500 words at a depth of 1, 500^2 words at a depth of 2, and 500^3 words at a depth of 3. Let W be the total number of words, which is $\sum_{i=1}^{500} 500^i$.

Before any phones have been heard, W words are in contention. After hearing the first phone,

$$\begin{aligned}
\frac{1}{500}W &= \frac{1}{500}\sum_{i=1}^{500} 500^i \\
&= \frac{500^1}{500} + \frac{500^2}{500} + \cdots + \frac{500^{500}}{500} \\
&= 1 + 500^1 + 500^2 + \cdots + 500^{499} \\
&= 1 + \sum_{i=1}^{499} 500^i \\
&= 1 + (W - 500^{500})
\end{aligned}$$

words are in contention. The constant 1 term is the root of the subtree that the first phone belongs to. Experiencing a second phone would remove the word belonging to the first phone from contention. As such, after hearing the second phone, the number of words still in contention is

$$\begin{aligned}
\frac{1}{500}\sum_{i=1}^{499} 500^i &= \frac{500^1}{500} + \frac{500^2}{500} + \cdots + \frac{500^{499}}{500} \\
&= 1 + 500^1 + 500^2 + \cdots + 500^{498} \\
&= 1 + \sum_{i=1}^{498} 500^i \\
&= 1 + (W - 500^{500} - 500^{499})\,.
\end{aligned}$$

This relationship between how many phones have been heard, n, and the number of words currently live, w_n, can be expressed as $w_n = f(n) = 1 + \sum_{i=1}^{500-n} 500^i$ for all $n \geq 1$ and $f(0) = W$. Note that the empty sum is taken to be 0, so that $f(500) = \sum_{i=1}^{500-500} 500^i = \sum_{i=1}^{0} = 0$. It is obvious that as n increases, the value of $f(n)$ decreases. This suggests that our desired upper bound such that $f(n)$ is $\mathcal{O}(\frac{1}{n})$ is possible and merely needs to be proved.

Proof. Let's begin by establishing a recurrence relation for the number of words live at any given phone n. Based on the above analysis, the recurrence relation is given as $T(n) = T(n-1) - 500^{501-n}$, with $T(0) = \sum_{i=1}^{500} 500^i$ and $T(1) = 1 + \sum_{i=1}^{500} 500^i - 500^{500}$. Now, we must prove that $T(n) \leq c\frac{1}{n}$ for all $n > 0$. That is, that $T(n)$ (and by extension $f(n)$) is $\mathcal{O}(\frac{1}{n})$. We will prove this statement by induction.

Base case

We must show the inequality holds for the base case $n = 1$. That is, $T(1) \leq c\frac{1}{n}$. To do so, let's choose a value for c, say, $\sum_{i=1}^{500} 500^i$. At $n = 1$, we have $T(1) = \sum_{i=1}^{500} 500^i - 500^{501-1} = \sum_{i=1}^{500} 500^i - 500^{500}$, and it is obvious that $\sum_{i=1}^{500} 500^i - 500^{500} \leq \sum_{i=1}^{500} 500^i$. Thus, we can establish that $T(1) \leq c\frac{1}{n}$ for some constant c, in this case, $\sum_{i=1}^{500} 500^i$.

Inductive step

Our recurrence relation is $T(n) = 1 + T(n-1) - 500^{501-n}$. What we wish to show now is that $T(n) \leq c\frac{1}{n}$ for all $n > 1$. If we assume that our statement holds true for all values less than n, we get $T(n-1) \leq c\frac{1}{n-1}$. We must now show this is true for $T(n)$. Because we assumed that $T(n-1) \leq c\frac{1}{n-1}$, it is implied that $T(n) = 1 + T(n-1) - 500^{501-n} \leq 1 + c\frac{1}{n-1} - 500^{501-n}$. We must now show that $1 + c\frac{1}{n-1} - 500^{501-n} \leq c\frac{1}{n}$. By algebraic manipulation, we can find a form where it is easier to show that this statement is true:

$$\begin{aligned} c\frac{1}{n-1} - 500^{501-n} &\leq c\frac{1}{n} \\ -500^{501-n} &\leq c(\frac{1}{n} - \frac{1}{n-1}) \\ -500^{501-n} &\leq c\frac{n-(n-1)}{n(n-1)} \\ -500^{501-n} &\leq c\frac{1}{n^2-n}\,. \end{aligned}$$

Because c is restricted by definition to be postitive, and our recurrence relation also restricts n such that $1 < n \leq 500$ (recall that $T(1)$ is defined separately, so we need not worry about the value of 1 in this instance), we can see that the last statement is true. That is, -500^{501-n} will always be negative, and $c\frac{1}{n^2-n}$ will always be positive, so it is clear that the inequality is true.

So long as the inequality holds at the base case $T(1)$, the inequality will hold generally. We know $T(1) = c - 500^{501-1} \leq 1 + c$ when $c = \sum_{i=1}^{500} 500^i$, for example. Thus, the base case holds.

Therefore, $T(n)$ is $\mathcal{O}(\frac{1}{n})$ for all $n \geq 1$. And, therefore, in a real language the number of words live at the n-th phone is $\mathcal{O}(\frac{1}{n})$ for all $n \geq 1$.

□

Because we do have a hard upper and lower bound for which $f(n)$ is defined, we can empirically verify this inductive proof as well. Using the Julia language, we can check that $f(n) \leq c\frac{1}{n}$ is true for every possible value of $n \geq 1$. The last line in the following code in Julia evaluates to `true`, corroborating the analytical proof.

```
W = sum(BigInt(500)^i for i in 1:500)
f = [1 + sum(BigInt(500)^i for i in 1:(500-n)) for n in 1:499]
push!(f, 1) # append the result of f(500) with the empty sum
O = [W * 1 / n for n in 1:500]
all(f .<= O) # check if all values of f <= the corresponding values of O
```

We can also verify that our recurrence relation matches our function f, where the last line of the following code in Julia evaluates to `true` as well.

```
W = sum(BigInt(500)^i for i in 1:500)
f = [1 + sum(BigInt(500)^i for i in 1:(500-n)) for n in 1:499]
push!(f, 1) # append the result of f(500) with the empty sum
T = [1 + W - BigInt(500)^(501-1)] # Start with the value of T(1)
for n in 2:500
  push!(T, last(T) - BigInt(500)^(501-n))
end
f == T
```

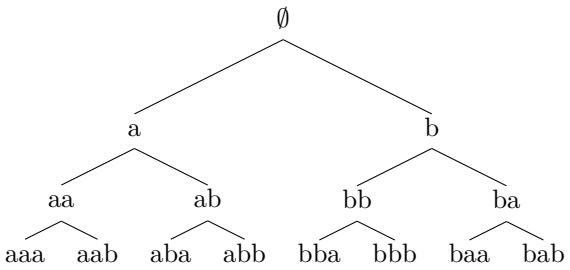
And, to really drive home just how much more complex this contrived language example is comparison to the same situation for human language, W, the number of possible words in this language, evaluates to a number that is approximately 3.06×10^{1349}. For reference, one googol is 1×10^{100}, and a million is merely 1×10^6. The number of atoms in the observable universe has been estimated (to within an order of magnitude) at 10^{80} (Eddington 1939) and 10^{78} (Silk 2005). Even if these estimates were one trillion times smaller than they needed to be, the number of atoms in the observable universe (let alone the prefix structure of words in a human language) would still be dwarfed by the number of words possible in this contrived language.

Lower bound

Now that an upper bound has been established, we must also show that $f(n)$ is $\mathcal{O}(\frac{1}{n})$ to prove Lemma 2. To perform this task, we will contrive another language example. This one will be uncontroversially simpler than a real language would be, and thus will provide some sort of lower bound, where we expect that the

nth phoneme in a real language will have as many or more words live as happens in this toy example.

Assume that the alphabet of this toy language consists of the symbols "a" and "b". Any combination of these symbols until a string length of three yields a valid word. The language can thus be expressed in the following prefix tree. Each node in the tree except for the root node represents a word.



After the first phone, 7 words are live. After the second phone, 3 words are live. And after the third phone, 1 word is live. This relationship is given by $2^{4-n} - 1$ words live after the $n - th$ phone. As a recurrence relation, this is given as $T(n) = T(n-1) - 2^{4-n}$. This is also a decreasing function, and we might want to try the simple $f(n) = \frac{1}{n}$ function first as a lower bound. That is, we wish to prove that $T(n)$ is $\Omega(\frac{1}{n})$.

Proof. By the substitution method, we must show that $T(n) \geq c\frac{1}{n}$. Because n can only take on three values, we can enumerate all of these possible values, and we have in fact already done so. If we choose $c = 1$, we must evaluate $7 \geq \frac{1}{1}$, $3 \geq \frac{1}{2}$, and $1 \geq \frac{1}{3}$. All three inequalities evaluate to true, so we can conclude that $T(n)$ is $\Omega(\frac{1}{n})$. Thus, for this small contrived language example, the number of words live at the n-th phone is $\Omega(\frac{1}{n})$ for all $n \geq 1$. Thus, because we chose this example to be a lower bound of what real language might do, the number of words $f(n)$ live at the n-th phone in a real language is $\Omega(\frac{1}{n})$ for all $n \geq 1$.

□

Synthesis

The proof for Theorem 1 is now at hand.

Proof. We have shown that for a case more complex than a real language might exhibit—which provides an upper bound on real language—the number of words that live at the n-th phone is $\mathcal{O}(\frac{1}{n})$. Similarly, for a case less complex than a real language—which provides a lower bound on real language—we have shown that the number of words live at the n-th phone is $\Omega(\frac{1}{n})$. Therefore, we can say that the number of words $w_n = f(n)$ live at the n-th phone in real human

language is $\Theta(\frac{1}{n})$. That is, Lemma 1 is true, and Lemma 2 is true, therefore, Theorem 1 is true.

□

Thus, we have proven that $w_n = f(n)$ is in a Zipfian relationship with n. That is, $f(n) \propto \frac{1}{n}$. And, this is precisely what we sought to prove. It is, thus, apparent that there is a Zipfian relationship between the number of phones encountered and the number of good candidates to be recognized during speech recognition, at least in cohort style models (Marslen-Wilson and Tyler 1980; Marslen-Wilson and Welsh 1978). Alternatively stated, the number of words that share a given prefix is in a Zipfian relationship with the number of phones in that prefix.

References

Cormen, Thomas H., Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2009. *Introduction to Algorithms*. Cambridge, MA, USA: MIT Press.

Eddington, Arthur Stanley. 1939. *The Philosophy of Physical Science: Tarner Lectures 1938*. Cambridge University Press.

Marslen-Wilson, William D, and Alan Welsh. 1978. "Processing Interactions and Lexical Access During Word Recognition in Continuous Speech." *Cognitive Psychology* 10 (1): 29–63. https://doi.org/10.1016/0010-0285(78)90018-X.

Marslen-Wilson, William, and Lorraine Komisarjevsky Tyler. 1980. "The Temporal Structure of Spoken Language Understanding." *Cognition* 8 (1): 1–71. https://doi.org/10.1016/0010-0277(80)90015-3.

Silk, Joseph. 2005. *On the Shores of the Unknown: A Short History of the Universe*. Cambridge University Press.

Zipf, George Kingsley. 1949. *Human Behavior and the Principle of Least Effort*. Cambridge, MA, USA: Addison-Wesley Press.