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_human_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_introduction_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 an n-ary tree with finite height D, the number of nodes in a subtree starting at depth $d \leq D$ is $\Theta(\frac{1}{d})$.*

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 $\mathcal{O}(\frac{1}{n})$.*

Lemma 3. *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 $\Omega(\frac{1}{n})$.*

General case

Consider a perfect n-ary tree with height D. "Perfect" in this context means that all nodes have n children, except those nodes at depth D, which all have no children. At depth 1, there is 1 node. At depth 2, there are n nodes. At depth 3, there are n^2 nodes. This pattern can be represented as function $g(d) = n^{d-1}$. The total number of nodes at or before depth d is, then, $s(d) = \sum_{i=0}^{d-1} n^i$. The number of nodes in a subtree starting at depth d would then be $f(d) = \sum_{i=0}^{D-d} n^i$. We wish to show that $f(d)$ is $\Theta(\frac{1}{d})$, which will prove Lemma 1.

Proof. We can choose constants $c_1 > 0$ and $c_2 > 0$ such that $c_1 \frac{1}{d} \leq f(d) \leq c_2 \frac{1}{d}$. By removing the common $\frac{1}{d}$ terms by multiplying by d, we can see that it is equivalent to finding c_1 and c_2 such that $c_1 \leq d \sum_{i=0}^{D-d} n^i \leq c_2$ for all values of d. Recall that n and D are ultimately constants for any particular tree, so $f(\cdot)$ is expressed only in terms of d. We know that the smallest value that $\sum_{i=0}^{D-d} n^i$ can be is 1 when $d = D$, so choosing $c_1 = 1$ satisfies the lower bound. We also know that the greatest value that $\sum_{i=0}^{D-d} n^i$ can be is $1 \sum_{i=0}^{D-1} n^i$ when $d = 1$. So, $c_2 = D \sum_{i=0}^{D} n^i$ satisfies the upper bound since this value is clearly larger than $1 \sum_{i=0}^{D-1} n^i$. Thus, $1 \leq d \sum_{i=0}^{D-d} n^i \leq D \sum_{i=0}^{D} n^i$ for all d on the interval $[1, D]$. Therefore, $f(d)$ is $\Theta(\frac{1}{d})$.

□

Note that we likely could have found tighter lower and upper bounds, but the asymptotic growth would still ultimatley be $\Theta(\frac{1}{n})$.

Upper bound on human language

Based on Lemma 1, we are now ready to prove Lemma 2. To contrive an example that is clearly more complex than human language, consider a perfect 500-ary tree with a height of 500. Framed in terms of language, consider that this contrived language 500 phonemes and that there are no phonotactic constraints on how to combine phonemes, up to a length of word length of 500. Also consider the prefix corresponding to an empty string, which we will not consider to be a word. The number of words w_n that share a given prefix of length n is expressed as $w_n = f(n) = \sum_{i=0}^{500-(n+1)} 500^i$. The offset as $n + 1$ reflects the fact that the depth is always one higher than the number of phonemes that have been heard. A small proof must be made that this situation is covered by the result from Lemma 1.

Proof. We proved in Lemma 1 that number of nodes in a subtree beginning at depth d is $\Theta(\frac{1}{d})$ for all d on the interval $[1, D]$. In the example here, we are considering the trees of depth d on the interval $[2, D]$ because the empty string resides in the node at depth $d = 1$. Because the interval $[2, D]$ is within the interval $[1, D]$, the result from Lemma 1 still applies, and we can state that the number of words w_n sharing a prefix of length n in the given example here is $\Theta(\frac{1}{n})$.

□

It should be obvious that this example is more complex than a human language, at least as regards the number of words that could be formed at each level of the tree. For example, there are 500 words of length 1, 500^2 words of length 2, etc. And, there are ultimately over 3.06×10^{1349} words possible in this example. 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_philosophy_1939] and 10^{78} [@silk_shores_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.

Because this example that provides an upper bound on the behavior of a real human language is $\Theta(\frac{1}{n})$, we can say that the number of words w_n sharing a prefix of length n in a human language is assuredly $\mathcal{O}(\frac{1}{n})$.

Lower bound on human language

It is now also possible to prove Lemma 3 based on Lemma 1. 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, except for the root node, which corresponds to an empty string. The tree structure of this example is visualized in Figure 1.

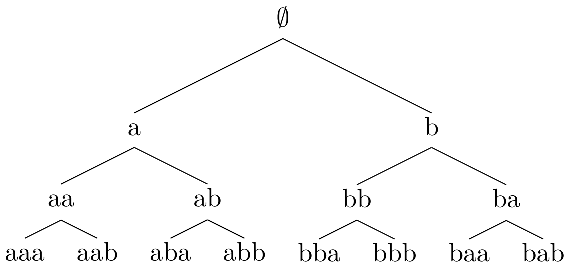

Figure 1: The tree structure for the contrived language example. Note that the root of the tree corresponds to the empty string or empty set, which is not considered to be a valid word. Every other node in the tree does correspond to a valid word in this language.

This example is a perfect binary tree with a height of 4. By analogy to the result from proving Lemma 2, we know that it matches the criteria for when Lemma 1 is true. As such, we know that the number of words sharing a prefix of length n is $\Theta(\frac{1}{n})$. The tree in this example is also obviously less complex than the situation in a real human language, so it provides a lower bound on human language. Thus, in a real human language, the number of words w_n sharing a prefix of length n is $\Omega(\frac{1}{n})$.

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 live at the n-th phone in real human language is $\Theta(\frac{1}{n})$. That is, Lemma 2 is true, and Lemma 3 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_temporal_1980; @marslen-wilson_processing_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